

Aus dem Zentrum der Radiologie  
des Klinikums der Johann Wolfgang Goethe-Universität  
Frankfurt am Main

(Direktor: Prof. Dr. Th. Vogl)

# Webbasierte Bildverteilung in der Radiologie: Performanzmessungen und Anforderungen an die Hardware

Dissertation  
zur Erlangung des Doktorgrades der Medizin  
des Fachbereichs Medizin  
der Johann Wolfgang Goethe-Universität

Vorgelegt von  
Michael Pietsch  
aus  
Frankfurt am Main

Frankfurt am Main, 2005



Promotionsbetreuer:  
Professor Dr. Th. Vogl  
Institut für Diagnostische und Interventionelle Radiologie





# 1 Inhaltsverzeichnis

<b>1</b>	<b>INHALTSVERZEICHNIS</b>	<b>1</b>
<b>2</b>	<b>EINLEITUNG</b>	<b>4</b>
2.1	Ziel dieser Untersuchung	4
2.2	Informations-Technologie (IT)-Systeme in der Radiologie	6
2.2.1	Radiologie-Informationssystem (RIS)	6
2.2.2	Digital Imaging and Communications in Medicine (DICOM)	6
2.2.3	Picture Archiving and Communication System (PACS)	8
2.2.4	Viewing Station	10
2.3	Der webbasierte Ansatz	10
2.4	Anwendungsszenarien	13
2.5	Grundlagen der Bildkompressionsverfahren	15
2.5.1	Grundlagen der JPEG Kompression	15
2.5.2	Grundlagen des Wavelet-Verfahrens	17
2.6	Java-Grundlagen	19
<b>3</b>	<b>MATERIAL UND METHODEN</b>	<b>21</b>
3.1	Studienaufbau	21
3.2	Server-Beschreibung	22
3.2.1	Server Hardware	22
3.2.2	Bildverteilungssoftware	22
3.3	Hochladende Rechner (Sender-PCs)	25
3.4	Anfordernde Rechner (Klient-PCs)	25
3.5	Test Objekte (Bilder)	26
3.5.1	Heruntergeladene Bilder	26
3.5.2	Hochgeladene Bilder	27
3.6	Entwicklung der Meßmethodik	28
3.6.1	Methode 1: Windows-Leistungsindikatoren	28
3.6.2	Methode 1b: Windows-Leistungsindikatoren, externe Protokollierung	30
3.6.3	Methode 2: Phototransistoren auf dem Bildschirm	31
3.6.4	Methode 3: Auslesen der Farbe von Bildschirmpunkten	34
3.7	Server Überwachungsanwendung	42

3.8	Vortests .....	42
3.8.1	Tests unserer Umgebung .....	42
3.8.2	Untersuchung des BVS .....	43
3.9	Testserien.....	43
3.9.1	Klient-Hardware-Modifikationen.....	43
3.9.2	Hochlade-Tests.....	45
3.9.3	Server-Modifikationen.....	47
<b>4</b>	<b>ERGEBNISSE.....</b>	<b>49</b>
4.1	Entwicklung der Meßmethodik .....	49
4.1.1	Qualität der Ergebnisse .....	49
4.1.2	Gebrauchsfähigkeit und Stabilität .....	51
4.2	Vortests .....	52
4.2.1	Tests unserer Umgebung .....	52
4.2.2	Untersuchung des BVS .....	54
4.3	Klient-Hardware-Tests.....	60
4.4	Hochlade-Tests .....	65
4.5	Server-Tests mit mehreren synchronisierten Klienten.....	67
4.5.1	Ohne Hochladen.....	67
4.5.2	Mit Hochladen.....	72
<b>5</b>	<b>DISKUSSION.....</b>	<b>76</b>
5.1	Grundsätzliches zum Studienaufbau.....	77
5.2	Entwicklung der Meßmethodik .....	78
5.2.1	Qualität der Ergebnisse .....	78
5.2.2	Gebrauchsfähigkeit und Stabilität .....	80
5.3	Vortests .....	83
5.3.1	Tests unserer Umgebung .....	83
5.3.2	Untersuchung des BVS .....	84
5.4	Klient-Hardware-Tests.....	86
5.5	Hochlade-Tests .....	89
5.5.1	Hochladekapazität .....	90
5.6	Server-Tests mit mehreren synchronisierten Klienten.....	92
5.6.1	Ohne Hochladen.....	92
5.6.2	Mit Hochladen.....	95

5.7	Optimierung der Leistung und der Ausfallsicherheit .....	97
<b>6</b>	<b>ZUSAMMENFASSUNG .....</b>	<b>100</b>
6.1	Meßmethodik.....	100
6.2	Messungen.....	100
<b>7</b>	<b>SUMMARY.....</b>	<b>102</b>
7.1	Methods .....	102
7.2	Results of measurements.....	102
<b>8</b>	<b>LITERATURVERZEICHNIS.....</b>	<b>104</b>
<b>9</b>	<b>TABELLEN .....</b>	<b>111</b>
9.1	Entwicklung der Meßmethodik .....	111
9.2	Klient-Hardware-Tests.....	111
9.3	Server-Tests mit mehreren synchronisierten Klienten.....	113
9.3.1	Ohne Hochladen.....	113
9.3.2	Mit Hochladen.....	114
<b>10</b>	<b>VERZEICHNISSE .....</b>	<b>115</b>
10.1	Abkürzungen .....	115
<b>11</b>	<b>INDEX .....</b>	<b>117</b>
<b>12</b>	<b>DANKSAGUNG .....</b>	<b>120</b>
<b>13</b>	<b>SCHRIFTLICHE ERKLÄRUNG UND PUBLIKATIONSVERZEICHNIS..</b>	<b>121</b>
<b>14</b>	<b>TABELLARISCHER LEBENS LAUF .....</b>	<b>122</b>

## 2 Einleitung

### 2.1 Ziel dieser Untersuchung

Die vorliegende Untersuchung hat sich zum Ziel gesetzt, ein vorhandenes, weit verbreitetes webbasiertes Bildverteilungssystem (BVS) auf dessen Einsatzfähigkeit hin zu untersuchen. Zu diesem Zweck wurden zunächst drei Methoden zur Messung von Lade- und Anzeigezeiten entwickelt und verglichen. Anschließend wurde in verschiedenen Meßreihen unter routineähnlichen Bedingungen untersucht, welche Anforderungen das untersuchte BVS an die einzelnen Klienten, das Netzwerk und den Bildverteilungsserver stellt.

Da die zu messende Zeitspanne sehr klein ist, in der Regel im Bereich weniger Sekunden, und da eine große Zahl an Wiederholungen nötig ist, ist es Menschen unmöglich, diese Messungen reproduzierbar und reliabel durchzuführen. Um die Genauigkeit durch Ausschluß dieser menschlichen Unzulänglichkeiten zu verbessern, ist ein automatischer Prozeß ideal. Dieser Prozeß darf aber keine Änderungen am BVS und dessen internen Abläufen voraussetzen, da sonst die Ergebnisse beeinflusst würden. Deshalb muß die Standard-Applikation ohne jede Änderung verwendet werden, der Meßvorgang muß auf die gleiche Weise mit der Standardapplikation interagieren, wie ein Mensch dies tun würde. Die drei entwickelten Methoden halten sich an dieses Paradigma.

Aufgrund der Komplexität von Bildverteilungssystemen ergibt sich eine große Zahl von Faktoren, die Übertragungs- und Anzeigezeiten beeinflussen können. Diese einflußnehmenden Faktoren können in zwei Gruppen eingeteilt werden, nämlich solche, die beim Testaufbau kontrolliert werden können, und solche, bei denen dies nicht möglich ist und die deshalb zufällige Abweichungen auslösen. In die erste Gruppe fällt unter anderem die Wahl des BVS, Konfiguration und Leistungsfähigkeit des Servers, Belastung des Servers, Leistungsfähigkeit und Anzahl der Klienten sowie die auf dem Server gespeicherte Datenmenge. Untersucht wurden Einflüsse durch die Hardware und das Betriebssystem der Klienten, durch verschiedene Server Hardware-Konfigurationen, durch verschiedene Kompressionsverfahren und durch gleichzeitiges Hochladen von Daten auf den Server.

Die zweite Gruppe der einflußnehmenden Faktoren umfaßt, ist aber nicht begrenzt auf Netzwerklast, unregelmäßige Anforderungen an den Server, Prozesse auf dem Server oder auf dem Klienten. Gerade in den großen Netzwerken einer Universitätsklinik kommt es vor, daß von Zeit zu Zeit große Datenmengen zwischen Servern übertragen werden, was durch Belastung des Netzwerkes Einfluß auf andere Übertragungsprozesse haben kann. Der Einfluß dieser Prozesse sollte allerdings in einem geschichteten Netz gering sein. Geschichtet bedeutet, daß aktive Netzwerkkomponenten die Datenpakete gezielt vom Sender an den Empfänger weiterleiten. Auf diese Weise können mehrere Punkt-zu-Punkt Verbindungen gleichzeitig mit voller Bandbreite arbeiten. Frühe, nicht-geschichtete Netzwerke sendeten grundsätzlich alle Datenpakete an alle Rechner, so daß ein großer Transfer von A nach B die Kommunikation zwischen C und D durch Belegung der Bandbreite bremste. Trotzdem gibt es in geschichteten Netzwerken gewisse Vorgänge, die auf sogenannten Broadcast-Paketen basieren. In diesem Fall wird ein Datenpaket ausnahmslos an alle Rechner verteilt, die mit dem Netzwerk verbunden sind.

Moderne Betriebssysteme wie UNIX, LINUX oder in unserem Fall Microsoft Windows (Microsoft® (MS), Redmond, WA, U.S.A) führen zu jedem Zeitpunkt mehrere Prozesse aus, die sich die Prozessorleistung des Systems teilen. Wichtige Prozesse in diesem Zusammenhang sind zum Beispiel Virens Scanner, der Windows Explorer, der die Benutzeroberfläche („Desktop“) darstellt, sowie Netzwerk-, Festplatten- und Grafikkartentreiber u.s.w. Alle diese Prozesse interagieren miteinander, reagieren auf externe Einflüsse wie Benutzereingaben oder Netzwerkpakete und führen in regelmäßigen Abständen gewisse Aufgaben aus. Dies führt dazu, daß in einem Rechner ständig Aktivität herrscht, die – insbesondere in Bezug auf gegenseitige Einflüsse – praktisch nicht vorhersagbar ist.

Um vergleichbare Ergebnisse liefern zu können, müssen alle Messungen aus diesen Gründen mehrfach durchgeführt und die Ergebnisse gemittelt werden.

## **2.2 Informations-Technologie (IT)-Systeme in der Radiologie**

### **2.2.1 Radiologie-Informationssystem (RIS)**

Der Einsatz von Computern im Krankenhaus ist historisch in drei verschiedenen Bereichen gewachsen. Bereits sehr früh wurden Computer in den klinischen Laboratorien eingesetzt, um die große Zahl der anfallenden Meßwerte zu erfassen und zu verarbeiten. Die zweite Gruppe, die sich dieser Technologie sehr aufgeschlossen zeigte, waren die radiologischen Abteilungen. Lange bevor an die digitale Verarbeitung von Bildern zu denken war, wurde die Technologie zur Befunderfassung, Patientenverwaltung und teilweise Terminplanung eingesetzt. Diese Aufgaben werden heute im sogenannten Radiologie-Informationssystem (RIS) abgebildet. Ein modernes RIS sollte neben diesen Aufgaben in der Lage sein, seine Patienten-Stammdaten mit dem Krankenhausinformationssystem abzugleichen und Befunde derart bereitzustellen, daß diese in ein Klinisches Arbeitsplatzsystem integriert werden können. Eine enge Anbindung an das Bildarchiv und die Übergabe von Patientendaten an die Modalitäten sollte selbstverständlich sein [38, 48, 49, 51, 74]. Darüber hinaus ist es zur Optimierung der Prozesse außerhalb der Radiologie sinnvoll, Anforderungen und Anmeldungen für Untersuchungen am Klinischen Arbeitsplatz System (KAS) durchführen und Bilder und Befunde innerhalb des KAS betrachten zu können [3, 50]. Ein RIS muß heute also in der Lage sein, mit vielen anderen externen Systemen im Krankenhaus zu kommunizieren.

### **2.2.2 Digital Imaging and Communications in Medicine (DICOM)**

Durch die Entwicklung der Computer Tomographie (CT) und Einführung weiterer digitaler Modalitäten in den 1970er Jahren wurde ein Vorgehen zum Bild- und Datenaustausch zwischen diesen Systemen nötig. Das American College of Radiology (ACR) und die amerikanische National Electrical Manufacturers Association (NEMA) entwickelten daraufhin einen Standard, der den Austausch unabhängig vom Modalitätenhersteller erlaubte und den Aufbau von Bildarchiven fördern sollte. Dieser wurde im Jahr 1985 in Version 1.0 veröffentlicht und seitdem weiterentwickelt (Version 2.0: 1988, Version 3.0: 1992).

Der heute als Digital Imaging and Communications in Medicine (DICOM) bekannte Standard sieht eine bestimmte Anzahl von Kommandos sowie

bestimmte Formate für die Daten vor. Zudem spezifiziert DICOM den Umgang mit dem Transmission Control Protocol / Internet Protocol (TCP/IP) (der ursprüngliche ACR-NEMA-Standard sah nur Punkt-zu-Punkt Verbindungen vor), legt ein Format für das langfristige Speichern von Daten fest und enthält Definitionen für Kommandos, die zwischen Geräten ausgetauscht werden können.

Der Standard legt jedoch nicht fest, wie die einzelnen Geräte intern aufgebaut sind oder was allgemein von einem Gerät erwartet werden kann. Dies erlaubt einzelnen Herstellern, für das jeweilige Anwendungsgebiet angepaßte Systeme zu schaffen, die über eine definierte Schnittstelle mit anderen Geräten kommunizieren. Allerdings legt der Standard auch keine Testprozedur fest, nach der die „DICOM Conformance“ geprüft werden kann, so daß es bis heute regelmäßig Probleme bei der Integration verschiedener Geräte und Systeme gibt, obwohl alle „DICOM konform“ sind [47].

Durch die Festlegungen des DICOM-Standards wird jedes Einzelbild fest mit Metainformationen, etwa dem Patientennamen, dem Aufnahmedatum, u.s.w. verbunden, was es zu einer austauschbaren Einheit macht. DICOM definiert folgende Begriffe:

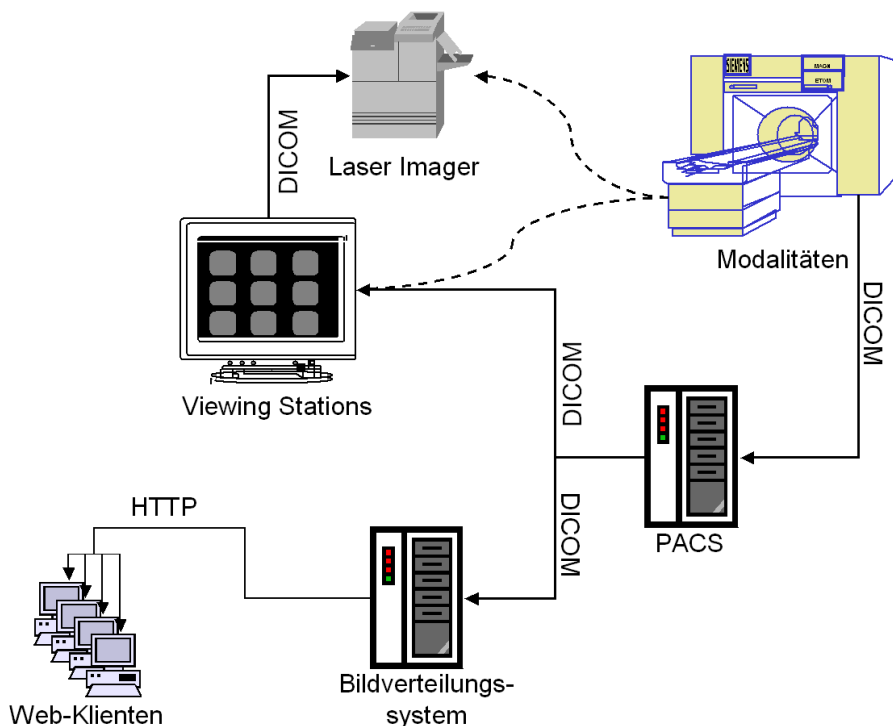
Eine Service Class kapselt verschiedene Funktionalitäten (Store, Print, Worklist, ...), die der DICOM-Standard beinhaltet. Ein Gerät muß nicht alle Service Classes unterstützen, um als „DICOM kompatibel“ bezeichnet zu werden. Die meisten Applikationen bzw. Geräte unterstützen nur jene Service Classes, die für ihren Verwendungszweck notwendig sind.

Ein Service Class Provider (SCP) ist ein Gerät bzw. eine Applikation, das eine bestimmte Service Class bereitstellt, während ein Service Class User (SCU) ein Gerät bzw. eine Applikation ist, das eine bestimmte Service Class nutzt.

Geräte, die die DICOM Storage Service Class unterstützen, erlauben das Versenden, Empfangen und Abspeichern von medizinischen Bildern. Hier wäre vor allem das Picture Archiving and Communication System (PACS) zu nennen.

### 2.2.3 Picture Archiving and Communication System (PACS)

Radiologische Abteilungen erstellen heute eine zunehmende Menge ihrer Bilder in digitaler Form. Traditionelle, filmgebundene Modalitäten werden zunehmend durch digitale Geräte ersetzt. Oftmals werden die aufgenommenen Studien jedoch sofort an einen Laserimager übermittelt und zur weiteren Verarbeitung ausgedruckt. Ein sogenanntes Picture Archiving and Communication System (PACS) stellt ein digitales Archiv dar, in dem Bilder im DICOM-Format abgelegt werden können. Ein PACS ist also gemäß DICOM-Spezifikation ein Storage Class Provider. Unter Verwendung des DICOM-Standards können Modalitäten und Viewing Stations Bilder an das PACS übermitteln und von dort abrufen.



**Abbildung 1: Bildfluß und Protokolle in der Radiologie. Gestrichelt: Traditionelle Vorgehensweise ohne PACS.**

Das klassische Vorgehen ohne PACS hat zweifelsohne den Vorteil der leichten Bedienbarkeit, des leichten Austausches der Bilder mit externen Institutionen und der Absturzsicherheit. Die Nachteile liegen jedoch ebenso auf der Hand: Ausgedruckte Bilder können zu jedem Zeitpunkt nur an einem Ort sein, sie müssen transportiert werden, vor dem Anschauen müssen sie gesucht werden, gehen durch falsches Einsortieren verloren und riesige Archive sind notwendig,



um die Aufbewahrungsfristen einzuhalten. Hinzu kommen Kosten für das Filmmaterial. Jeder dieser Nachteile bindet in großem Umfang Ressourcen, ist aber nicht in jedem Fall finanziell meßbar. Demgegenüber sind die Kosten für die Einführung eines PACS relativ leicht zu berechnen. Wesentliche Kostenfaktoren sind dabei die Archivierung der Bilddaten und die Infrastruktur zur Betrachtung der Bilder innerhalb und außerhalb der Radiologie. Ratib et al. berichteten jedoch bereits im Jahr 2000, daß die Einführung eines PACS zwar kurzfristig höhere Kosten bedeutet, langfristig aber höhere Produktivität bei gleichbleibenden Kosten bringt [57]. Insbesondere wird innerhalb und außerhalb der Radiologie der Workflow wesentlich verbessert, was vor allem den Klinikern zu Gute kommt [3, 27, 51, 66]. Allerdings muß darauf hingewiesen werden, daß die Einsparungen und Vorteile durch die Einführung eines PACS mit Bildverteilung vor allem außerhalb der Radiologie liegen.

Oben wurde bereits kurz erwähnt, daß die höchsten Kosten durch die Notwendigkeit zur langfristigen Archivierung der Bilddaten entstehen, wobei auf die langfristige Verfügbarkeit der Bilder ebenso geachtet werden muß (gemäß § 28 RöV in der Fassung vom 30. April 2003) wie darauf, daß die Daten nicht verändert oder von Fremden eingesehen werden können. Neue Daten, und solche, auf die häufig zugegriffen wird, werden in der Regel auf Festplatten „online“ gespeichert. Für die langfristige Speicherung haben sich verschiedene Konzepte entwickelt. Manche Institutionen lagern auch die langfristigen Daten auf Festplatten online, während andere Digital Versatile Disk (DVD)-, write-once read-many (WORM) oder Bandlaufwerke oder Speicherroboter verwenden [13, 22, 63, 64, 68, 72, 73]. Bei der Wahl des Speicherverfahrens ist zu beachten, daß die Menge der Daten in den kommenden Jahren durch Mehrzeilen-CTs, 3D-Rekonstruktionen und andere neue Technologien weiter stark zunehmen wird [22, 68]. Gleichzeitig ist mit einem weiteren Preisverfall bei Speichermedien zu rechnen, der diese zunehmende Datenmenge kompensieren wird. Man kann jedoch nicht davon ausgehen, daß, wie in vielen Bereichen der Informationstechnologie, die zunehmende Rechenleistung und fallende Speicherpreise zukünftig alles billiger machen [22, 63, 68].

Entscheidend beim Aufbau einer PACS-Lösung ist die Integration des Systems mit dem RIS. RIS und PACS können zwar unabhängig betrieben werden, doch

ergeben sich erst mit der Integration die entscheidenden Vorteile wie Workflow-Optimierung, Einsparen doppelten Eintippens und sichere Zuordnung der Daten und Befunde [38, 48, 49, 51, 74].

#### **2.2.4 Viewing Station**

In den letzten Jahren hat die Anzahl der produktiv betriebenen PACS-Installationen stetig zugenommen. Um dem Ziel einer klinikumsweiten Bild- und Befundverteilung näher zu kommen, wurden verschiedene Wege beschritten. Der traditionelle Ansatz ist die Installation einer großen Anzahl DICOM-Workstations, welche entweder direkt auf das zentrale PACS der Radiologie oder auf eigene Arbeitsgruppenserver zugreifen und über unterschiedliche Schnittstellen die Befunde aus dem RIS extrahieren. Solche Viewing Stations verwenden in der Regel große und besonders kontrast- und leuchtkraftstarke CRT-Monitore sowie spezialisierte Software und zeigen Bilder und Befunde an [15, 21, 25, 27, 60]. Eine Integration in eine elektronische Patientenakte und Anbindung an das Klinische Arbeitsplatzsystem (KAS), die beide am Arbeitsplatz des Stationsarztes installiert sind, ist jedoch meist nicht vorgesehen.

Da dies nicht zuletzt, aufgrund teurer Workstations und hoher Lizenzkosten, ein relativ kostenintensives Unterfangen ist, wurde mehrfach versucht, mit Hilfe der Internettechnologie Lösungen zu verwirklichen, welche in einfacher Weise den gestellten Anforderungen gerecht werden sollen. Dazu gehört vor allem der webbasierte Zugang zu den radiologischen Informationsdatenbanken, wie dem RIS und dem PACS. Die in den Datenbanken vorhandenen Informationen können dabei sowohl Nutzern innerhalb eines lokalen Netzwerkes als auch zuweisenden Ärzten oder kooperierenden Krankenhäusern außerhalb des Klinikums zur Verfügung gestellt werden.

### **2.3 Der webbasierte Ansatz**

Webbasierte Verfahren sind solche, die auf der Technologie des World Wide Web aufbauen, das seinerseits auf Technologien des Internets aufbaut. Die wichtigsten Errungenschaften des World Wide Web sind der Browser und der Link. Der Browser ist eine Software, die von einem Server mittels des standardisierten Hypertext Transfer Protocol (HTTP) Daten abholt und diese anzeigt. Dabei ist es unerheblich, welches Betriebssystem zum Einsatz kommt.

In der vorliegenden Untersuchung wurde der weitverbreitete Microsoft Internet Explorer (MSIE) (Microsoft® (MS), Redmond, WA, U.S.A) als Browser eingesetzt. Der Link erlaubt es dem Benutzer, durch einen einzigen Mausklick auf ein Textsegment oder ein Bild ein anderes Dokument anzufordern. Browser und Link erlauben die leichte Navigation durch Verknüpfung verschiedener Dokumente und Listen. Ein einfacher Link kann Daten auf verschiedenen Servern und aus verschiedenen logischen Systemen verbinden und somit vormals getrennte Systeme verknüpfen.

Es gibt zahllose Möglichkeiten, diese Errungenschaften der Internettechnologie für klinische Anwendungen in einem Krankenhaus oder einer Röntgenabteilung einzusetzen. Die für die Radiologie bedeutsamste ist die Bild- und Befundverteilung.

Der Zugriff auf die textbasierten Daten und Befunde, die im RIS gespeichert sind, läßt sich mit den webbasierten Technologien leicht realisieren. Aus technischer Sicht bestehen zwischen diesen Informationen und dem Inhalt gewöhnlicher Internetseiten keine Unterschiede. Einzig der Datenschutz muß in besonderer Weise berücksichtigt werden.

Systeme zur Wiedergabe von radiologischen Bildern hingegen stellen besondere Ansprüche. Radiologische Bilder unterscheiden sich durch viel höhere Auflösung und Farbtiefe deutlich von Bildern, die man gemeinhin im Internet findet. Die größere Auflösung und Farbtiefe haben zur Folge, daß radiologische Bilder eine im Internet unübliche Datenmenge umfassen. Zudem sollte der Betrachter in der Lage sein, die Bilder zu manipulieren, das heißt zu fenstern und zu zoomen, was mit regulären Bildern im Internet weder gefordert noch möglich ist.

Systeme zur webbasierten Bildverteilung lassen sich in zwei Kategorien einteilen. Solche, die Bilddaten in Internet-Standardformaten wie Joint Picture Experts Group (JPEG), Tagged Image File Format (TIFF) oder Graphics Interchange Format (GIF) übertragen und solche, die spezialisierte Formate und Betrachtungssoftware verwenden. Der Vorteil der Internet-Standardformate liegt darin, daß außer dem Browser keine besondere Software benötigt wird. Der Nachteil liegt darin, daß eine Interaktion mit dem Bild nicht möglich ist. Der

Benutzer muß immer mit dem Bild auskommen, das ihm vom Server zur Verfügung gestellt wird. Bestenfalls kann ein anders voreingestelltes (window/level/zoom) Bild beim Server angefordert werden. Eine solche Anforderung braucht immer Zeit, flüssiges Arbeiten ist deshalb selbst in schnellen Netzen nicht möglich.

In der Gruppe der Systeme mit spezialisierter Betrachtungssoftware kommen in der Regel Browser-Plugins [20, 76] oder Java-Applets [4, 23, 28, 36, 39, 71] zum Einsatz. Plugins sind Programme, die sich nach der Installation im Browser einklinken und ihren Dienst tun. Eine Installation ist bei Plugins immer nötig. Im Gegensatz dazu bieten Java-Applets den Vorteil, daß sie nicht installiert werden müssen. Sie werden zusammen mit der Internetseite heruntergeladen und auf dem Rechner des Benutzers lokal ausgeführt. Sowohl Plugins als auch Java-Applets haben den Vorteil, daß klientenseitig Programmcode ausgeführt werden kann, der Funktionalität bereitstellt, die ein normaler Browser nicht kennt. Im vorliegenden Fall kann dieser Programmcode heruntergeladene Bilddaten interpretieren und anzeigen, ohne daß diese in einem Internet-Standardformat vorliegen. Eine flüssige Interaktion mit den Daten ist möglich, da sie komplett auf dem Klient zur Verfügung stehen und nicht bei jeder Änderung vollständig neu vom Server angefordert werden müssen.

Systeme, die mit Java-Applets oder Plugins arbeiten, lassen sich wiederum in zwei Kategorien einteilen. Solche, die reine DICOM-Daten übertragen [20, 76] und solche, die vorverarbeitete, zumeist komprimierte Bilder an den Klient senden [28, 35, 41, 50, 59, 61]. Reine DICOM-Daten haben den entscheidenden Nachteil, daß zur Darstellung eines Bildes eine große Datenmenge nötig ist, die immer komplett übertragen werden muß, bevor das Bild angezeigt werden kann. DICOM-Bilder haben hingegen den Vorzug, daß nicht unbedingt ein spezialisierter Server oder eine spezielle Server-Software nötig ist und daß sie für die Verteilung nicht vorbereitet oder umgewandelt werden müssen. Die Vorteile einer Vorverarbeitung der Bilder überwiegen jedoch bei weitem. Vorverarbeitete Bilder kommen mit deutlich kleineren Datenmengen aus und können zum Teil bereits angezeigt werden, bevor alle Daten auf dem Klienten angekommen sind. Bereits während der Übertragung kann ein unscharfes Bild

dargestellt werden, das bereits eine grobe Orientierung erlaubt. In vielen Fällen ist es, aufgrund der relativ geringen Auflösung eines regulären Personal Computer (PC) Monitors, nicht nötig, die kompletten Bilddaten zu übermitteln, da diese ohnehin nicht angezeigt werden können. Der Download der Daten wird dann entsprechend kürzer dauern. Sollte der Benutzer sich entscheiden, eine Bildregion zu vergrößern, so können Bilddaten beim Server nachgefordert werden.

## **2.4 Anwendungsszenarien**

Aufgrund der beschriebenen Vorteile des webbasierten Ansatzes eignet sich dieser insbesondere für eine Reihe von Anwendungsszenarien, die mit reinen DICOM-Workstations nur schwer oder gar nicht realisierbar sind und deshalb zum Teil nicht im großen Stil in Angriff genommen wurden. Es lassen sich fünf Grundszenarien abgrenzen.

Das bedeutendste Szenario ist die Bild- und Befundverteilung. Diese läßt sich weiter aufteilen in zwei verschiedene Zweige. Am häufigsten eingesetzt wird sie bisher zur Bildverteilung innerhalb einer Institution, dieses Szenario ist auch am besten beschrieben und untersucht [1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15, 17, 18, 19, 20, 23, 28, 29, 30, 32, 36, 38, 40, 41, 43, 44, 45, 50, 51, 53, 56, 60, 63, 65, 71, 76]. Der kostenintensive traditionelle Ansatz wäre, an ausgewählten Orten in der Institution DICOM-Workstations einzurichten. Hier hat die Webtechnologie den entscheidenden Vorteil, daß alle vorhandenen PCs im Arztzimmer oder Standard-PCs im Besprechungsraum [70] oder im OP [7] oder zukünftig gar mobile Endgeräte für das Betrachten von Röntgenbildern eingesetzt werden können und daß eine Installation nicht notwendig ist. Im Gegensatz zu Viewing Stations verringert dies Ausfallzeiten, verhindert Schlängengebilde zu Stoßzeiten und dient einem einheitlichen stationären Workflow. Gleichzeitig können Bilder und Befunde nahtlos in das KAS und die elektronische Patientenakte eingebunden werden. Gerade diese Anbindung stellt einige Anforderungen an KAS, RIS und Bildverteilungssystem (BVS) und wird heute in besonderem Maße verfolgt [3, 16, 19, 50, 74].

Die Webtechnologie erlaubt es darüber hinaus erstmals, Bilder und Befunde auch einweisenden Ärzten oder anderen Institutionen zur Verfügung zu stellen,

die nicht an das Local Area Network (LAN) und die IT-Infrastruktur des Krankenhauses angebunden sind, sondern über ein Wide Area Network (WAN) oder das Internet auf die Daten zugreifen. Daraus ergeben sich völlig neue Nutzungsszenarien wie Bildverteilung innerhalb von Forschungsprojekten [14, 52] und Bildverteilung an Niedergelassene oder Externe Organisationen [31, 52, 56, 69, 74]. Somit ist webbasierte Bildverteilung im Rahmen einer einheitlichen Digitalen Gesundheitsakte interessant. Es ist, aufgrund von Bandbreitenlimitationen und des begrenzten Speicherplatzes eines Praxis-PC, nicht möglich, niedergelassene Ärzte oder Einweiser mit DICOM-Bildern zu beschicken. Im Rahmen der Einführung der Digitalen Gesundheitskarte werden jedoch alle Arztpraxen in Deutschland mit voraussichtlich breitbandigen Internetanschlüssen versorgt. Browser sind ohnehin weitgehend installiert oder sind für den Umgang mit der Digitalen Gesundheitskarte nötig. Damit sind die Grundvoraussetzungen der Niedergelassenen für das webbasierte Betrachten von Röntgenbildern erfüllt.

Ein weiteres Grundszenario ist die Teleradiologie. Der zunehmende Kostendruck im Gesundheitswesen führt in Deutschland zunehmend zu einer Konsolidierung. Es zeichnet sich deshalb ab, daß zukünftig nicht zu jeder Tages- und Nachtzeit an jeder Institution ein ausgebildeter Radiologe verfügbar sein wird. Andere Länder verwenden in dünn besiedelten Gebieten bereits seit längerem routinemäßig Teleradiologiesysteme. Die webbasierte Bildverteilung erlaubt es, im Rahmen einer Telekonferenz schnell eine zweite Meinung einzuholen oder gar die Befundung vollständig aus der Ferne durchzuführen [8, 35, 39, 40, 53, 69]. Neben dem webbasierten Ansatz wird allerdings derzeit vor allem in Entwicklungsländern versucht, mit Hilfe von Digitalkameras die Bilder abzufotografieren oder mit einfachen Flachbettscannern einzulesen und zu verschicken [54, 62].

Das vierte Grundszenario stellen Referenzbilddatenbanken dar [58]. Verschiedene Projekte zum Aufbau von Referenzbilddatenbanken im Internet wurden über die Jahre begonnen. Da praktisch jeder mit dem Internet verbundene PC die Voraussetzungen für das webbasierte Betrachten von Bildern erfüllt, ist diesen Vorhaben einiger Erfolg vorauszusagen. Um eine Interaktion (window/level/zoom) zu erlauben, ist auch hier zu fordern, daß Java-Applets

oder Plugins zum Einsatz kommen. Andernfalls werden nur nicht-interaktive Bilder verteilt, die oftmals noch nicht einmal ein Heranzoomen erlauben, was den Wert der Referenzbilder deutlich herabsetzt.

Das fünfte Grundszenario ist die Aus- und Weiterbildung. Im Internet können auf der Basis von webbasierten Bildverteilungssystemen Lehrgänge für alle Bereiche angeboten werden. Niedergelassene und Studenten haben gleichermaßen alle Möglichkeiten des Zugriffs auf Bilddaten und können so online an Kursen teilnehmen oder diese vor- und nachbereiten [58, 67, 75].

In allen fünf Szenarien sind durch die Prinzipien der webbasierten Bildverteilung große Fortschritte möglich.

## 2.5 Grundlagen der Bildkompressionsverfahren

### 2.5.1 Grundlagen der JPEG Kompression

Der Joint Picture Experts Group (JPEG)-Kompression liegt die diskrete Kosinustransformation (DCT) und deren Gegenstück, der inversen DCT zugrunde [33, 55]. Die DCT ist eine lineare, orthogonale Transformation, welche ähnlich der Diskreten Fouriertransformation (DFT) ein zeitdiskretes Signal vom Ortsbereich in den Frequenzbereich transformiert.

$$S(v,u) = \frac{C(v)}{2} \frac{C(u)}{2} \sum_{y=0}^7 \sum_{x=0}^7 s(y,x) \cos \frac{(2x+1)u\mathbf{p}}{16} \cos \frac{(2y+1)v\mathbf{p}}{16}$$

$$s(y,x) = \sum_{v=0}^7 \frac{C(v)}{2} \sum_{u=0}^7 \frac{C(u)}{2} S(v,u) \cos \frac{(2x+1)u\mathbf{p}}{16} \cos \frac{(2y+1)v\mathbf{p}}{16}$$

**Abbildung 2: Diskrete 2D-Kosinustransformation (oben) und inverse DCT (unten)**

$s(x, y)$  ist der 2D-Zahlenwert, während  $S(v, u)$  den entsprechenden 2D-DCT-Koeffizienten bezeichnet.  $x$  und  $y$  bezeichnen den horizontalen bzw. vertikalen Versatz und  $u$  und  $v$  die horizontalen bzw. vertikalen Frequenzen. Des weiteren gilt:

$$C(u) = \begin{cases} 1/\sqrt{2} & \text{für } u = 0 \\ 1 & \text{für } u < 0 \end{cases} \quad C(v) = \begin{cases} 1/\sqrt{2} & \text{für } v = 0 \\ 1 & \text{für } v < 0 \end{cases}$$

Zunächst wird die zu komprimierende Bitmap in Segmente zu 8x8 Pixeln aufgeteilt. Daraus ergibt sich ein Feld aus 8x8 Grauwerten:

59	58	59	58	57	123	135	136
61	61	61	60	59	125	137	134
62	61	60	60	120	132	135	137
62	61	60	60	124	134	133	133
61	60	60	62	126	131	135	132
61	60	61	122	130	137	133	131
61	60	60	123	132	133	135	134
60	60	60	125	138	137	135	132

**Tabelle 1: Grauwerte  $s(x, y)$  innerhalb einer 8x8-Matrix.**

Auf die Farb- oder Grauwerte  $s(x, y)$  dieser Pixel wird nun die DCT angewendet, woraus sich eine Matrix von 8x8 DCT-Koeffizienten  $S(v, u)$  ergibt. Zu diesem Zeitpunkt ist noch keine Reduktion der Datenmenge entstanden, im Gegenteil ist die Anzahl der nötigen Bits sogar gestiegen, da mit 64-bit Fließkommazahlen gearbeitet wird. Wird die inverse DCT auf diese Daten angewendet, so ergeben sich wieder die Rohdaten.

775.500	-252.185	1.759	55.258	2.500	-5.092	-4.246	-7.917
-55.344	0.246	73.151	-5.874	-47.822	-0.745	23.458	6.233
-4.791	4.487	4.536	-21.136	-1.558	27.992	-0.925	-29.744
-7.193	3.639	4.420	-9.836	-3.119	13.579	1.221	-13.133
-9.000	-2.452	8.166	6.493	-6.500	-9.268	3.382	11.048
6.018	-1.524	-7.118	1.619	7.589	-3.746	-1.824	1.811
10.644	-1.797	-13.925	3.198	9.304	-3.397	-2.536	0.580
-1.832	-3.727	3.967	4.574	0.130	-9.217	-0.626	12.336

**Tabelle 2: DCT-Koeffizienten  $S(v, u)$  der Matrix aus Tabelle 1**

Die Reduktion der Daten wird nun zum einen durch Rundung, zum anderen durch die Verwendung einer Quantisierungstabelle  $Q(v, u)$  erreicht.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

**Tabelle 3: Quantisierungstabelle  $Q(v, u)$**

Im Quantisierungsschritt werden die Einzelwerte durch ganzzahlige Werte ersetzt, woraus sich die Matrix  $T(v, u)$  ergibt.



$$T(v,u) = \frac{S(v,u)}{Q(v,u)}, \quad v,u = 1,2,\dots,7$$

Das Runden auf ganzzahlige Werte hat einen Informationsverlust zur Folge, so daß die Rohdaten  $s(x, y)$  auch nach Anwenden der inversen DCT nicht mehr wiederhergestellt werden können. Durch Anwendung einfacher Kompressionsverfahren kann die Matrix  $T(v, u)$  der Zahlenwerte stark komprimiert werden, da sie abhängig von der verwendeten Quantisierungstabelle  $Q(v, u)$  eine große Zahl von Nullen enthält.

48	-23	0	3	0	0	0	0
-5	0	5	0	-2	0	0	0
0	0	0	-1	0	0	0	-1
-1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

**Tabelle 4: Matrix  $T(v, u)$  unter Berücksichtigung der DCT-Koeffizienten aus Tabelle 2 und der Quantisierungstabelle (Tabelle 3)**

Die Quantisierungstabelle wurden empirisch ermittelt. Für bestimmte Zwecke können angepaßte Quantisierungstabellen entwickelt werden. Auch die Größe der Matrix von 8x8 Pixeln ist das Ergebnis empirischer Untersuchungen.

Daraus ergibt sich, daß eine JPEG-Kompression immer verlustbehaftet ist. Aus rechtlichen Gründen kann sie somit nicht zur Befundung verwendet werden.

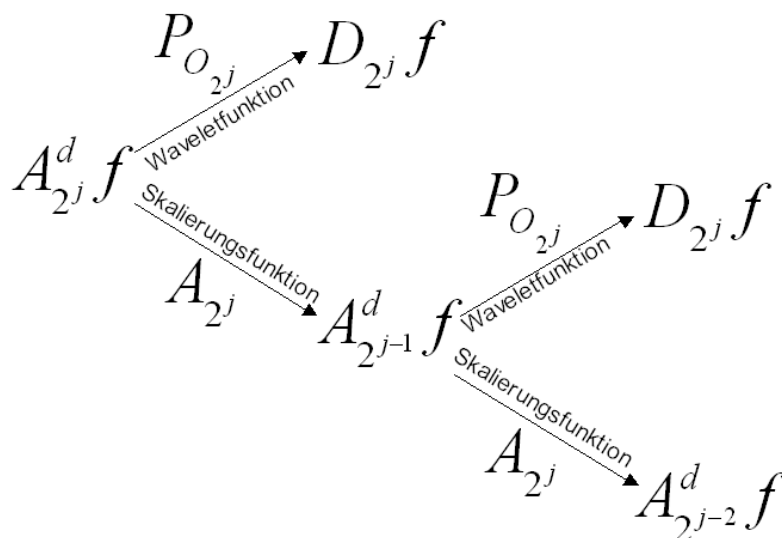
### 2.5.2 Grundlagen des Wavelet-Verfahrens

Unter Wavelets versteht man Methoden der Signalverarbeitung und Kompression, deren erster Schritt die schnelle Wavelet-Transformation beinhaltet [34]. Wavelets wurden zu Beginn der 1990er Jahre als Durchbruch in der Bild- und Tonkompression propagiert, finden jedoch bis heute in der Breite keine Anwendung. Unter anderem macht sich der noch immer kaum verbreitete JPEG2000-Standard [61, 62] dieses Verfahren zu nutze.

Wavelets basieren darauf, daß mittels geeigneter Wavelet- und Skalierungsfunktionen eine Hochpaß- und eine Tiefpaß-Filterung durchgeführt wird. Daraus ergibt sich noch keine Kompression, jedoch können die resultierenden

Komponenten zum Teil vernachlässigt werden, was zur Verringerung der Datenmenge führt. Der vernachlässigte Teil der Daten kann gespeichert und später nachgeladen werden, um zu einer verlustfreien Wiedergabe zu gelangen.

Jedes elektronische Gerät, darunter auch Modalitäten in der Radiologie, können ein Signal nur mit einer bestimmten endlichen Abtastrate erfassen. Diese Auflösung kann man als  $2^0 = 1$  setzen,  $A_1^d f$  ist dann die diskrete Approximation des Signals bei dieser Auflösung. Aus  $A_1^d f$  können mit Hilfe von Skalierungsfunktionen mit dem Operator  $A_{2^j}$  alle niedrigeren Auflösungen  $A_{2^j}^d f$  für  $j < 0$  errechnet werden. Dies entspricht einer Reduktion der Zahl der Abtastpunkte, oder einer Tiefpaßfilterung. Gleichzeitig wird mit Hilfe einer Wavelet-Funktion mit dem Operator  $P_{O_{2^j}}$  die Differenz  $D_{2^j} f$  zwischen den Originaldaten und den aus der Skalierungsfunktion resultierenden Informationen ermittelt. Aus dem Ergebnis der Skalierungsfunktion und dem Ergebnis der Wavelet-Funktion können dann die Originaldaten rekonstruiert werden [42].



**Abbildung 3: Aufteilung der Bildinformationen durch Skalierungs- und Wavelet-funktionen. Das Verfahren kann beliebig weitergeführt werden.**

Somit ist das Wavelet-Verfahren strenggenommen kein Kompressionsverfahren, sondern dient nur dazu, das ursprüngliche Signal so in gefilterte Komponenten zu zerlegen, daß Daten intelligent weggelassen werden können.

Gleichzeitig ermöglicht das Verfahren die verlustfreie Speicherung von Signalen oder Bilddaten.

Im Rahmen der Übermittlung radiologischer Bilder ist es oftmals ausreichend, eine skalierte Version des Bildes anzuzeigen, da das Anzeigegerät, aufgrund einer Größenbeschränkung des Bildschirms, die Daten ohnehin nicht vollständig und in voller Auflösung darstellen kann. Vergrößert der Benutzer einen Ausschnitt, so können weitere Daten nachgeladen werden. Zudem ergibt sich aus dem oben gesagten, daß das Wavelet-Verfahren sich vor allem dazu eignet, schnell ein unscharfes, verlustbehaftetes Bild darzustellen, das durch das Nachladen weiterer Daten an Schärfe gewinnt [41, 59].

## **2.6 Java-Grundlagen**

Java ist eine objektorientierte Programmiersprache, die an C++ angelehnt ist und Anfang der 1990er Jahre von der Firma Sun Microsystems (Santa Clara, CA, U.S.A.) entwickelt wurde. Ziel war ursprünglich die Entwicklung eines einfachen Betriebssystems für Microcontroller, wie sie in Kaffeemaschinen, Waschmaschinen, Telefonanlagen u.s.w. vorkommen. Als in den folgenden Jahren das Internet und das WWW aufkamen, versuchte man die Sprache, die konzeptionell plattformunabhängig ist, in diesem Bereich einzusetzen. Kleine Java-Programme, sogenannte Applets, können in Internetseiten eingebunden werden, um für Interaktivität zu sorgen.

Grundlage der Plattformunabhängigkeit von Java ist die Verwendung einer sogenannten Virtual Machine (VM). Übliche Anwendungen werden in einer Programmiersprache entwickelt und anschließend kompiliert, wobei die Anweisungen der Programmiersprache direkt in Befehle für den Prozessor des Zielsystems umgewandelt werden. Damit ist eine einmal kompilierte klassische Anwendung auf einen Prozessortyp und ein Betriebssystem festgelegt, also plattformgebunden. Im Gegensatz dazu wandelt der Java-Compiler die Befehle der Programmiersprache zunächst in sogenannte Bytecodes um. Diese Bytecodes werden bei Ausführung des Programms von der sogenannten Java VM verstanden, interpretiert, d. h. in Prozessorbefehle umgewandelt und an den Prozessor weitergegeben. Die VM wiederum ist eine standardisierte Umgebung, die gegenüber dem Java-Programm ein einheitliches Umfeld

simuliert, während sie selbst für den Einsatz auf einer bestimmten Plattform entwickelt wurde. Derartige VMs existieren für verschiedene Plattformen und dienen somit gewissermaßen als Adapter zwischen Java-Programm und einzelner Plattform.

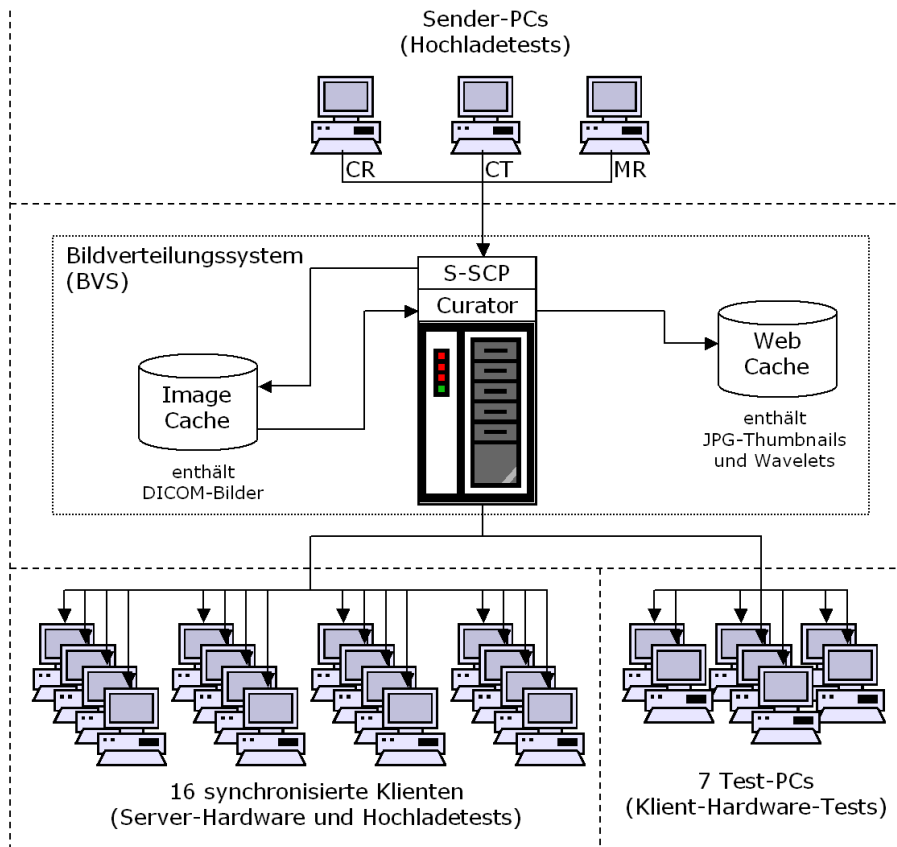
Aus diesem Vorgehen ergeben sich einige Vor- und einige Nachteile. Durch die Notwendigkeit zur Interpretation des Bytecodes im Rahmen der Ausführung des Programms leidet die Arbeitsgeschwindigkeit. Zudem bringt die plattform-unabhängige Entwicklung einige schwer lösbare Probleme mit sich, da beispielsweise Pfadangaben für den Dateizugriff unter UNIX anders realisiert werden als unter Windows, ähnliches gilt auch für Bildschirmausgaben, wobei hier insbesondere die Größe von Steuerelementen verschieden ausfällt. Verschiedene Hersteller haben im Laufe der Zeit verschiedene Versionen von VMs entwickelt und dabei die Vorschriften der Standards verschieden interpretiert. Aus diesem Grund kommt es entgegen den Erwartungen immer wieder zu Inkompatibilitäten zwischen einzelnen VMs und Software muß in der Regel für eine bestimmte VM entwickelt und getestet werden. Damit fordern Java-Programme oftmals die Existenz einer bestimmten VM, die ggf. nur für eine bestimmte Plattform verfügbar ist.

Die Vorteile liegen vor allem in der Plattformunabhängigkeit selbst, die Entwicklungszeit beim Erstellen von Programmen spart. Darüber hinaus hat das Konzept der VM den Vorteil, daß Java-Programme in einem abgeschlossenen Raum ablaufen und somit nur begrenzten Schaden anrichten können. Alle Zugriffe, die über diesen Raum hinausgehen, können sehr leicht durch entsprechende Sicherheitsfunktionen kontrolliert werden. Damit kann sichergestellt werden, daß aus dem Internet heruntergeladene Java-Programme keine Daten löschen oder modifizieren können, keinen Zugriff auf die Hardware erhalten und sich nicht wie Viren vervielfältigen können.

Ein entscheidender Vorteil für Anwender und Systemadministratoren ist, daß Java-Programme in der Regel, abgesehen von der Existenz einer VM, keine besonderen Bedingungen an die Umgebung stellen, in der sie laufen sollen. Damit entfällt die Installation ebenso wie unerwünschte Interaktionen zwischen verschiedenen Programmen.

## 3 Material und Methoden

### 3.1 Studienaufbau



**Abbildung 4: Testaufbau.** Drei Sender-PCs wurden jeweils mit Bildern einer bestimmten Modalität bestückt, die im Rahmen der Hochladetests an den Server übertragen wurden. Der S-SCP nimmt die Daten an und speichert sie im „Image Cache“. Der Curator wandelt sie in Wavelets und JPEGs um, die im „Web Cache“ abgelegt werden. Im Rahmen der Server-Tests forderten 16 Klienten synchron Bilder beim Server an. Unabhängig davon wurden im Rahmen der Klient-Hardware-Tests 7 Klienten mit dem Server verbunden.

Um die verschiedenen Aufgaben eines BVS nachstellen und messen zu können, wurde ein Testaufbau gewählt, in dem alle Aspekte einer Bildverteilung nachempfunden werden konnten (Abbildung 4). Die Funktion der Modalitäten bzw. des PACS, nämlich das Senden von Bildern an den Bildverteilungsserver, wurde von der Software eFilm (Version 1.8.1; eFilm Medical Inc, Toronto, Kanada) auf drei Sender-PCs übernommen, die nach Modalität sortiert die Bilder im DICOM-Format an das BVS übertragen. Die anfordernden Rechner wurden durch 16 identische Klient-PCs simuliert, die automatisiert Bilder vom Bildverteilungsserver abholten, anzeigten und die dafür benötigte Zeit maßen.

Im Rahmen der Klient-Hardware-Tests wurden die sieben Test-PCs nacheinander einzeln an das Netzwerk angeschlossen. Zur Messung der Zeiten und zur Automatisierung der Abläufe wurde auf den Sender-PCs, auf dem Bildverteilungsserver und auf den Klient-PCs jeweils zusätzliche Software ausgeführt, die an entsprechender Stelle beschrieben wird.

## **3.2 Server-Beschreibung**

### **3.2.1 Server Hardware**

Als Standard-Bildverteilungsserver wurde ein Primergy 470 (Fujitsu Siemens Computers) Server mit zwei Pentium III 700 MHz Prozessoren und 1 Gigabyte (GB) Random Access Memory (RAM) verwendet. Zwei Gruppen von Festplatten wurden via Small Computer System Interface (SCSI) und einem Redundant Array of Independent Disks (RAID) Controller mit 16 Megabyte (MB) Cache angebunden. Das Betriebssystem und die Bildverteilungssoftware wurden auf zwei gespiegelten 9 GB Festplatten (RAID-I) installiert, die Bilder und Daten selbst fanden auf drei als RAID-V (Stripeset mit Parity) konfigurierten 18 GB Festplatten Platz. Die Verbindung mit dem Local Area Network (LAN) wurde durch zwei FastEtherExpress pro/100+ Netzwerkkarten hergestellt. Diese wurden zu einem FastEtherChannel (Cisco Technology, San Jose, CA, U.S.A.) zusammengeschaltet, so daß sich eine Gesamtbandbreite von 200 Mbit/s ergab. Dabei ist jedoch anzumerken, daß in einem FastEtherChannel pro Richtung nur je 100 Mbit/s zur Verfügung stehen. Allerdings können Daten in beiden Richtungen gleichzeitig mit dieser Geschwindigkeit übertragen werden, während sich bei der Verwendung einzelner 100Mbit/s-Netzwerkkarten für beide Richtungen zusammen 100 Mbit/s zur Verfügung stehen. Der Anschluß erfolgte an einen Catalyst WS-C 6509 (Cisco Technology, San Jose, CA, U.S.A.) Backbone Switch. Die Bildverteilungssoftware Mitra Exhibit setzte als Betriebssystem Windows NT Server 4.0, Service Pack (SP) 6 (Microsoft® (MS), Redmond, WA, U.S.A) voraus, das auf diesem Rechner eingesetzt wurde.

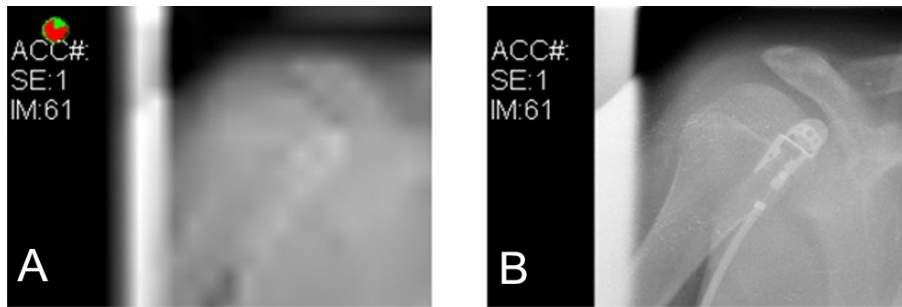
### **3.2.2 Bildverteilungssoftware**

Als Bildverteilungssoftware wurde Mitra Exhibit gewählt, das ursprünglich von der Firma Mitra Imaging Inc. (Waterloo, Ontario, Canada) entwickelt wurde. Exhibit wird mit leichten Modifikationen unter einer Reihe von verschiedenen

Namen durch Firmen wie AGFA (Mortsel, Belgien) (Web 1000), Philips (Eindhoven, Niederlande) (Easy Web) und wie in unserem Fall Kodak (Rochester, NY, U.S.A.) (WebAccess) vertrieben. Es speichert alle Bildinformationen mit Ausnahme der Bilder selbst in einer SQL-Server 7.0 Datenbank (Microsoft®, Redmond, WA, U.S.A.), die Bilder werden im Dateisystem abgelegt. Nach dem Import via DICOM [46] werden die Bilder in ein proprietäres Wavelet-Format [41, 42, 59] umgewandelt.

Beim Hochladen werden die Bilder vom BVS zunächst vom Storage Service Class Providers (S-SCP) angenommen und auf der Festplatte im Verzeichnis „Image Cache“ abgelegt. Nachdem eine Studie vollständig empfangen wurde, beginnt der Curator-Prozeß damit, die Bilder entsprechend der auf dem Server vorgegebenen Einstellung in verlustfreie oder verlustbehaftete wavelet-komprimierte [42, 59] Bilder umzuwandeln (Primärkompression) und für jedes Bild eine Miniaturansicht im JPEG-Format [55] anzulegen. Dieser Schritt wird als Nachbearbeitung bezeichnet. Die Wavelet- und Miniaturansicht-JPEG-Dateien werden im Verzeichnis „Web Cache“ abgelegt (Abbildung 4, Seite 21). Ist verlustbehaftete Primärkompression auf dem Server eingestellt, so können die Bilder später nicht mehr in verlustfreier Qualität angezeigt werden, da während dieser Umwandlung Bildinformation unwiederbringlich verloren geht.

Das Anzeigen der Bilder erfolgt mit Hilfe eines Java-Applets (BVS-Klient-Applet), das den Microsoft Internet Explorer (MSIE) und die Microsoft Virtual Machine (VM) voraussetzt. Dieses Applet erlaubt nach der Anmeldung am Server die Auswahl eines Patienten und einer Studie und zeigt Bilder an. Dabei ist es unter anderem möglich, Bilder in 1 x 1 bis 4 x 4-Tableaus anzuzeigen und mit den Bildern zu interagieren, sie zu zoomen, zu fenstern u.s.w. Neben den von uns verwendeten 1 x 1 und 4 x 4-Tableaus ist es möglich, verschiedene Bilder nebeneinander zu betrachten, andere Tableau-Größen zu wählen (2 x 2, 3 x 3), Bildserien in schneller Folge als „Film“ wiederzugeben u.s.w.



**Abbildung 5:** Nach dem Eintreffen der ersten Daten wird vom BVS-Klient-Applet bereits ein unscharfes Bild angezeigt (A), das schärfer wird (B). Links oben ist das Fortschrittsicon zu erkennen.

Bei der Anzeige der Bilder macht das Exhibit Klient-Applet Gebrauch von der Möglichkeit des Wavelet-Formates, verschiedene Frequenzen dynamisch nachzuladen [42, 59] (Siehe auch Grundlagen des Wavelet-Verfahrens, Seite 17). Beim Herunterladen werden Bilder zunächst unscharf angezeigt, beim Eintreffen weiterer Daten werden die Bilder schärfer (Abbildung 5). Dabei wird immer nur die Menge an Daten heruntergeladen, die zur Anzeige eines scharfen Bildes bei der entsprechenden Bildschirmauflösung nötig ist. Wird ein anderer Bildausschnitt gewählt oder gezoomt, so werden dynamisch Daten nachgeladen. Jeder Ladevorgang wird von einem kleinen Icon symbolisiert, das in der linken oberen Ecke des Anzeigebildschirms erscheint und verschwindet, sobald alle Daten eingetroffen sind und das Bild scharf dargestellt wird.

Das System erlaubt es, wenn verlustfreie Primärkompression auf dem Server eingestellt ist, mit Hilfe eines Buttons in einen verlustbehafteten Anzeigemodus umzuschalten, um Netzwerkbandbreite zu sparen. In diesem Fall komprimiert der Server die angeforderten Bilder erst direkt vor dem Versand (Sekundärkompression). Die Anzeige eines verlustfrei komprimierten Bildes ist nur dann möglich, wenn die Bilder nicht schon beim Import via DICOM einer verlustbehafteten Primärkompression unterzogen wurden. Wurde eine verlustbehaftete Primärkompression vorgenommen, hat die Auswahl des verlustbehafteten Anzeigemodus auf dem Klient-PC keine Auswirkung, da ohnehin nur verlustbehaftete Bilder angezeigt werden können. Eine zusätzliche Kompression wird nicht vorgenommen.



### **3.3 Hochladende Rechner (Sender-PCs)**

Wie oben beschrieben, wurden drei Sender-PCs konfiguriert, die anstelle von Modalitäten oder dem PACS Bilder an unseren Exhibit-Server sendeten. Diese waren mit 1.7 GHz Pentium IV Prozessoren und 256 MB RAM ausgestattet und von Fujitsu Siemens Computers hergestellt. Netzwerkkarte und Grafikkarte waren direkt auf dem Mainboard integriert. Alle drei wurden mit 100 Mbit/s im Full Duplex Modus an einen Catalyst WS-X 2948G, Cisco<sup>®</sup> Workgroup Switch (Cisco Technology, San Jose, CA, U.S.A.) angebunden, der via Gigabit-link mit dem Server Backbone Switch verbunden war. Als Betriebssystem kam Windows 2000 Professional zum Einsatz, Internet Explorer war installiert. Darüber hinaus wurde der DICOM Viewer eFilm (Version 1.8.1; eFilm Medical Inc, Toronto, Canada) als DICOM Storage Service Class User (S-SCU) [46] installiert. Das Freeware-Tool Automouse (Kakuya Yamamoto, Japan) wurde eingesetzt, um mit Hilfe von Scripts während der Tests automatisch und kontinuierlich Bilder aus eFilm heraus an den Bildverteilungsserver zu senden.

### **3.4 Anfordernde Rechner (Klient-PCs)**

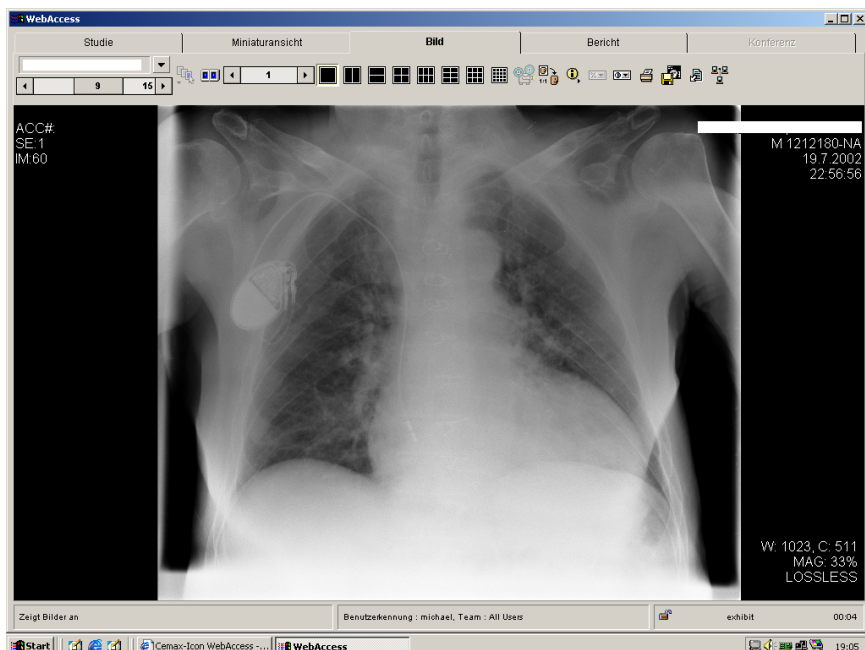
Die 16 anfordernden Rechner waren Fujitsu Siemens PCs mit 1 GHz Pentium III Prozessoren, 256 MB RAM und 20 GB Festplatten. Die Grafikkarte (INTEL 82815, 4 MB) und die Netzwerkkarten (INTEL Pro/100 VM) waren direkt auf dem Mainboard installiert. Alle wurden mit 100 Mbit/s im Full Duplex Modus an den Workgroup Switch (Catalyst WS-X 2948G, Cisco Technology, San Jose, CA, U.S.A.) angebunden, der wiederum über ein Gigabit-link mit dem Server Backbone Switch verbunden war. Die anfordernden PCs verwendeten Windows 2000 Professional als Betriebssystem und Internet Explorer mit der Java Virtual Machine (beide Microsoft<sup>®</sup> und Version 5.0) als Grundlage für das BVS-Klient-Applet um Bilder anzuzeigen.

Darüber hinaus wurde während der Tests ein Testkoordinator-PC eingesetzt, der als Betriebssystem Windows 2000 Advanced Server einsetzte.

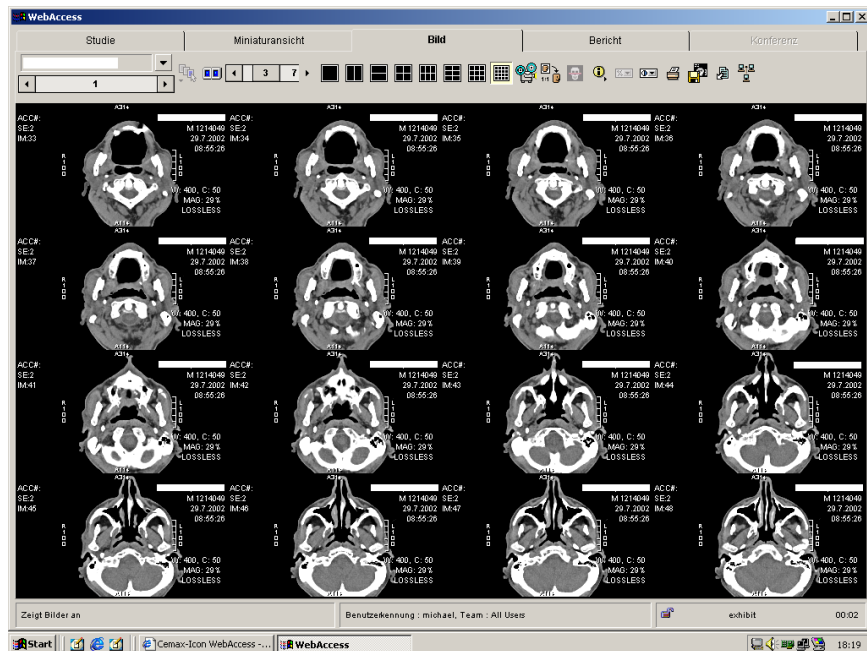
## 3.5 Test Objekte (Bilder)

### 3.5.1 Heruntergeladene Bilder

Für die Messung der Anzeigezeiten im Rahmen der Klient- und Servertests wurden drei Bildtypen definiert, die den häufigsten Modalitäten entsprechen (Computed Radiography (CR), Computer Tomography (CT) und Magnetic Resonance Imaging (MR)). Die Gruppe der CR-Bilder enthielt vier Thoraxaufnahmen und eine Abdomenaufnahme. CR-Bilder wurden bei den Versuchen jeweils im Vollbild als 1 x 1 dargestellt (Abbildung 6). CT- und MR-Studien wurden immer in einer 4 x 4 Matrix dargestellt, so daß auf einem Bildschirm 16 Bilder mit geringer Auflösung dargestellt wurden (Abbildung 7). Die Gruppe der CT-Bilder setzte sich aus vier abdominellen und einem kranialen CT zusammen. Für MR-Studien wurden drei abdominelle MRs und zwei Lungen-MRs gewählt. Die Charakteristika der Bilder entsprachen üblichen Untersuchungstechniken. Dabei hatten die CR-Bilder im DICOM-Format eine durchschnittliche Größe von 8 MB pro Bild, die CT-Bilder 0,5 MB pro Bild im DICOM-Format (entspricht 8 MB für 16 Bilder) und die MR-Bilder 0,125 MB pro Bild im DICOM-Format (entspricht 2 MB für 16 Bilder).



**Abbildung 6: Darstellung eines CR-Bildes in Einzeldarstellung unter Verwendung des Mitra BVS-Klient-Applet. Verlustfreie Kompression.**



**Abbildung 7: Darstellung von 16 CT-Bildern in einem 4 x 4-Tableau unter Verwendung des Mitra BVS-Klient-Applet. Verlustfreie Kompression.**

Um sicherzustellen, daß die während der Tests automatisch ausgewählten Patienten bei zunehmender Zahl von Bildern auf dem Server nicht ihre Position in der Auswahlliste änderten und somit den Ablauf der Scripts störten, mußten im Rahmen der Messungen bei gleichzeitigem Upload andere Bildserien verwendet werden. Diese entsprachen in den Charakteristika den oben genannten Bildern, die Namen der entsprechenden Patienten begannen mit A und B, so daß diese immer an oberster Stelle in der Auswahlliste verblieben.

### 3.5.2 Hochgeladene Bilder

Im Rahmen der Upload-Tests wurden die drei Sender-PCs jeweils mit Bildern einer Modalität bestückt. Der erste Sender-PC enthielt 745 CR-Bilder, entsprechend 5.648 MB an Bilddaten. Der zweite wurde mit 13.014 CT-Bildern gefüllt, die insgesamt eine Größe von 6.541 MB belegten. Der dritte Sender-PC enthielt 19.728 MR-Bilder, die 7.353 MB an Daten umfaßten. Die Bilder auf diesen PCs wurden zufällig aus den auf dem PACS der Uniklinik Frankfurt vorhandenen Bildern und Studien ausgewählt, so daß sich eine der Routine entsprechende Verteilung der untersuchten Körperregionen ergab.

## **3.6 Entwicklung der Meßmethodik**

Um das Anzeigeverhalten von webbasierten Bildverteilungssystemen zu messen, kann eine Reihe von Methoden angewendet werden. Die meisten Autoren [4, 5, 6, 23, 43] verlassen sich bei ihren Verfahren auf menschliche Tester und handelsübliche Stoppuhren. Dies hat den Vorteil, daß die Messungen ohne großen technischen Aufwand durchgeführt werden können, zugleich aber den Nachteil, daß die Zahl der jeweils durchgeführten Messungen und deren Genauigkeit sehr eingeschränkt sind.

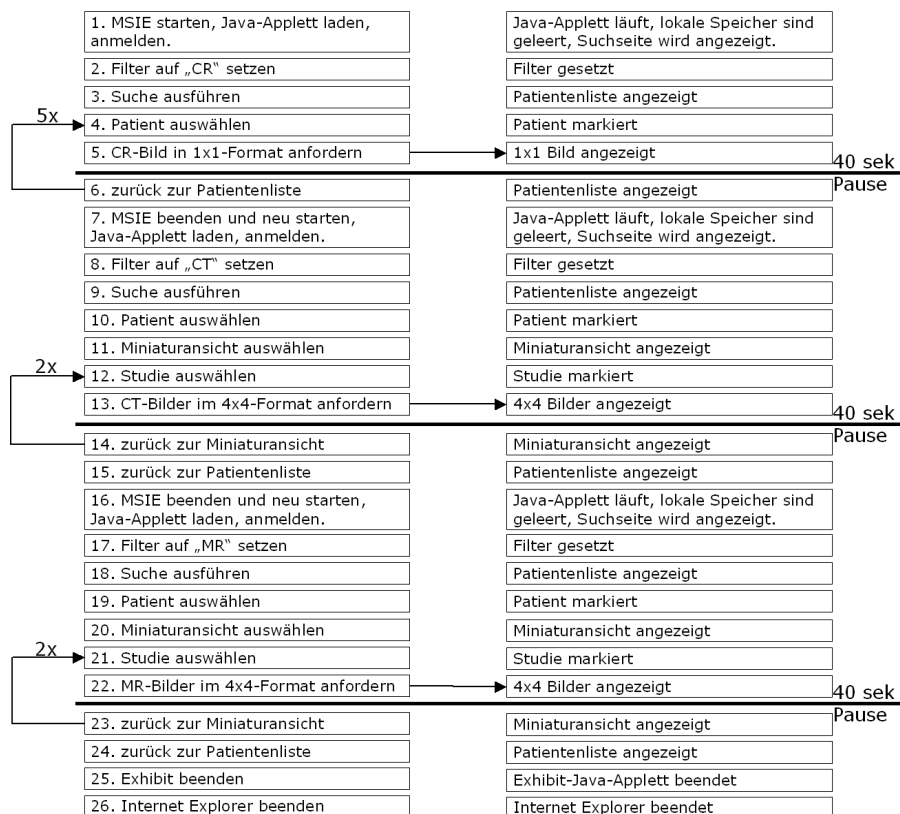
Aus diesem Grund setzten wir es uns zum Ziel, ein automatisiertes Meßverfahren zu definieren, das einfach und automatisierbar sein und zugleich reliable und reproduzierbare Ergebnisse liefern sollte. Dabei sollte das Verfahren das bestehende Bildverteilungssystem in keiner Weise beeinflussen und ein genaues Verständnis der Abläufe innerhalb des Bildverteilungssystems sollte nicht von Nöten sein („Closed Box Testing“). Drei Methoden wurden entwickelt und deren Ergebnisse miteinander verglichen.

### **3.6.1 Methode 1: Windows-Leistungsindikatoren**

Die erste Methode basierte auf den Leistungsindikatoren (performance-counters), die vom Systemmonitor (perfmon) von Windows 2000 und Windows NT bereitgestellt werden.

Mit Hilfe des Freeware-Tools Automouse wurden Scripts entwickelt, die automatisch und kontinuierlich Bilder vom Webserver abriefen und so einen menschlichen Benutzer simulierten. Automouse ist in der Lage, bestimmte Bildausschnitte auf dem Bildschirm zu suchen und in einer vorgegebenen Reihenfolge Mausclicks darauf auszuführen bzw. Tastatureingaben zu simulieren. Mit einem einfachen Script-Editor können die zu suchenden Bildschirmausschnitte und die darauf auszuführenden Aktionen sowie Wartezeiten definiert werden. Dabei sucht Automouse den Bildschirm tatsächlich nach den entsprechenden Ausschnitten ab. Diese Suche ist von verschiedenen Faktoren abhängig, so daß sie in manchen Fällen einen kürzeren Zeitraum beansprucht als in anderen. Ein einmal so erstelltes Script ist jedoch sehr robust.

Das Script meldete sich zunächst am BVS an, rief eine Liste von Patienten ab, wählte im Anschluß zunächst einen Patienten und dann ein CR-Bild bzw. eine CT- oder MR-Studie aus und rief diese ab. Dabei wurden fünf CR-Einzelbilder sowie je zwei CR- und MR-Serien von zwei Patienten heruntergeladen. Zwischen den verschiedenen Bildtypen wurde das BVS-Klient-Applet und der Internet Explorer beendet und neu gestartet, um die Bilder aus dem lokalen Speicher zu entfernen und später die Ergebnisse leichter zuordnen zu können. Die genauen Schritte können Abbildung 8 entnommen werden.



**Abbildung 8: Schematischer Ablauf der Messungen mit Methode 1.** In der linken Spalte sind die Aktionen aufgeführt, rechts die jeweiligen Ergebnisse. Verbindungspfeile symbolisieren Zeitmessungen. Links sind die Wiederholungen durch Pfeile symbolisiert. Zwischen den einzelnen Schritten wurden mehrere lange Pausen eingebaut, eingezeichnet sind nur die Pausen während des Bildladevorganges.

Darüber hinaus wurde eine kleine Anwendung in MS Visual Basic 6.0 SP4 (Microsoft® (MS), Redmond, WA, U.S.A) entwickelt, die mit hoher Geschwindigkeit bestimmte Leistungsindikatoren in eine Semikolon-getrennte (Colon Separated Value, CSV) Datei schrieb. Die mit Windows 2000 ausgelieferte Version von perfmon kann diese Daten nur im Ein-Sekundentakt protokollieren,

während die mit Windows NT gelieferte Version zwar eine höhere Sampling-Rate unterstützt, jedoch nur Gruppen von Leistungsindikatoren protokollieren kann, was zu sehr schnell wachsenden Protokolldateien mit hoher Belastung des beobachteten Systems führt.

Für die vorliegende Messung wurden folgende Indikatoren gewählt: „Process(Idle)\ % Processor Time“ gab die nicht benötigte Central Processing Unit (CPU)-Zeit an, „Process(AUTOMOUSE)\ % Processor Time“ erlaubte Rückschlüsse auf die Aktivität von Automouse und wurde vor allem für grobe Schätzungen verwendet aber nicht weiter analysiert. Der wichtigste Indikator war „Process(IEXPLORE)\ % Processor Time“, der die Aktivität des Internet Explorers angab und damit auch die Aktivität des innerhalb dieses Prozesses laufenden BVS-Klient-Applets.

Die so gewonnenen CSV-Dateien wurden durch ein automatisches System analysiert, in dem Grenzwerte für alle Indikatoren definiert wurden, aus denen im Anschluß die Dauer einzelner Aktivitäten und somit einzelner Vorgänge errechnet werden konnte. Anstiege in der Aktivität der verschiedenen Prozesse konnten allerdings auch durch andere Vorgänge, die mit dem Anzeigen von Bildern nichts zu tun haben, ausgelöst werden. Zudem war der Zeitraum zwischen Start des Tests und einem Peak variabel, da Automouse wie oben erwähnt die Bildschirmausschnitte, auf die es reagiert, auf dem Bildschirm sucht. Einzige Anhaltspunkte für die Zuordnung eines Peaks zu einer Aktion waren somit die Zahl der vorangegangenen Peaks und die Länge des fraglichen Peaks in Relation zu anderen Peaks der gleichen Klasse. Aufgrund der großen Zahl der ermittelten Meßwerte mußten die Ergebnisse anhand einiger Regeln automatisch geprüft und anschließend manuell bestätigt werden.

### **3.6.2 Methode 1b: Windows-Leistungsindikatoren, externe Protokollierung**

Das Prinzip von Methode 1 wurde so modifiziert, daß die CSV-Dateien nicht auf dem BVS-Klient-Computer gespeichert, sondern über das Netzwerk von einem externen Computer aus ausgelesen wurden. Dies entlastete den BVS-Klient-Computer von Schreibaarbeit, produzierte jedoch eine zusätzliche Netzwerkbelastung von rund 400 kB/sec.

### 3.6.3 Methode 2: Phototransistoren auf dem Bildschirm

Die zweite Methode basierte auf drei Phototransistoren, die direkt auf dem Bildschirm des BVS-Klient-Computers geklebt wurden und dort Helligkeit maßen. Somit konnte unter Ausschluß jeder Beeinflussung des BVS-Klient-Computers gemessen werden. Die Phototransistoren waren mit BC 237 Transistoren verbunden, die das Signal verstärkten, bevor es durch einen TLC 549 Analog-Digital-Konverter (ADC) 256-stufig digitalisiert wurde. Das Ergebnis dieser Messung wurde über die parallele Schnittstelle eines zusätzlichen Rechners ausgelesen (Abbildung 9). Auf diesem PC lief das Betriebssystem MS Disk Operating System (DOS) Version 6.22 und eine kleine in MS Quick-Basic 4.5 (beide Microsoft®, Redmond, WA, U.S.A) entwickelte Anwendung, die die Meßwerte in eine Datei eintrug. Neben den Phototransistoren wurde ein Reed-Relais an die parallele Schnittstelle angeschlossen, das den linken Mausknopf der Maus des BVS-Klient-PCs überbrückte. So konnte der externe Rechner durch einen simulierten Mausklick Ladevorgänge starten. Um die Anmeldung am BVS vorzunehmen und ein Bild auszuwählen, wurde wie in Methode 1 Automouse verwendet. Wie bei Methode 1 waren lange Pausen notwendig um sicherzustellen, daß keine Desynchronisation zwischen dem externen PC, Automouse und dem BVS-Klient-Applet eintrat.

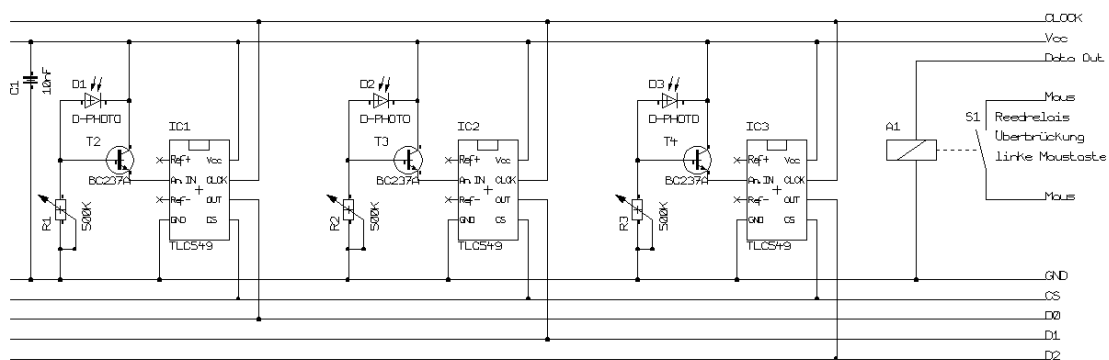
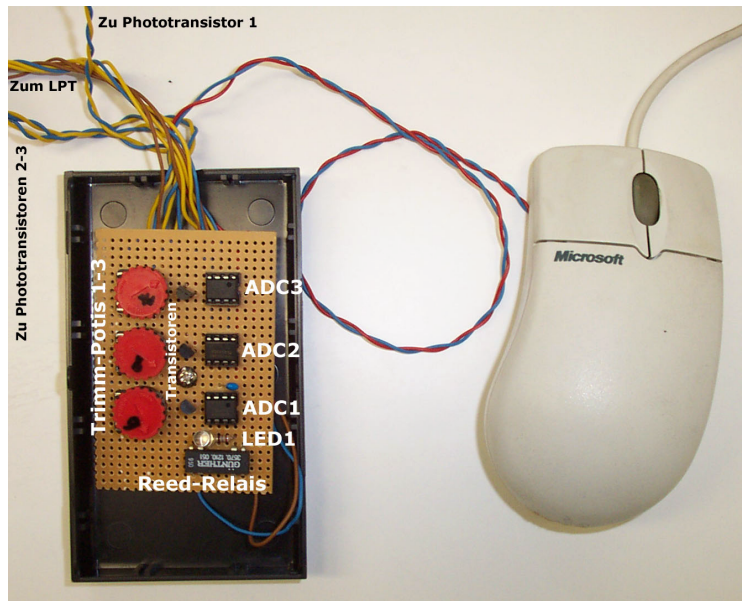


Abbildung 9: Schaltplan der Meßelektronik für Methode 2.



**Abbildung 10: Selbstgebaute Meßelektronik für Methode 2. Klar zu erkennen sind die Analog-Digital-Konverter (ADC 1 - 3), das Relais und die Verbindung zur Maus.**

Die drei Phototransistoren wurden so auf dem Bildschirm platziert, daß die gemessenen Bereiche vor, während und nach Abschluß eines Ladevorganges deutliche Helligkeitsunterschiede aufwiesen. Der erste wurde auf einen Bereich geklebt, der im Bildanzeige-Modus des BVS-Klient-Applets schwarz wurde. Dieser Bereich lag außerhalb des Bildes selbst, so daß festgestellt werden konnte, ob tatsächlich ein Ladevorgang gestartet wurde. Der zweite klebte in der Mitte des Bildes, und wurde durch die Anzeige des Mediastinums einer Thoraxaufnahme beleuchtet. Der dritte Phototransistor schließlich wurde an der Stelle angebracht, an der das BVS-Klient-Applet eine kleine Uhr einblendet, um zu signalisieren, daß ein Ladevorgang läuft (Abbildung 11). Dieses Icon verschwindet nach Abschluß des Ladevorganges (Abbildung 5, Seite 24). Die genauen Schritte können Abbildung 12 entnommen werden.



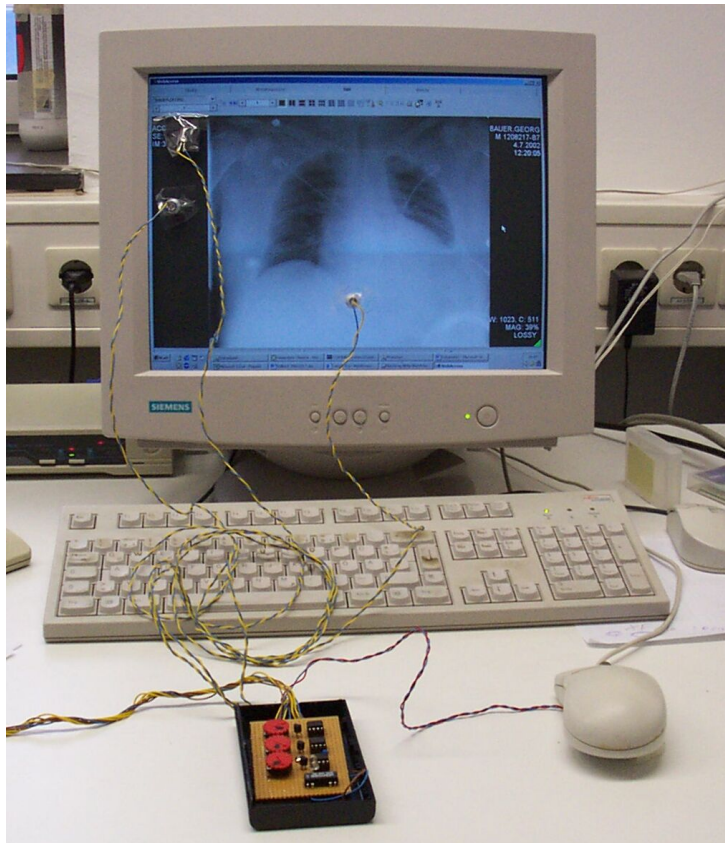


Abbildung 11: Test-PC mit Phototransistoren, Meßelektronik und überbrückter Maus.

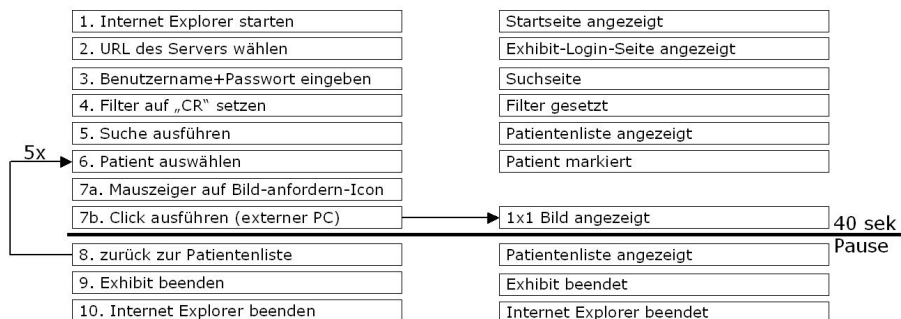


Abbildung 12: Schematischer Ablauf der Messungen mit Methode 2. In der linken Spalte sind die Aktionen aufgeführt, rechts die jeweiligen Ergebnisse. Der Verbindungspfeil symbolisiert die Zeitmessungen. Links sind die Wiederholungen durch Pfeile symbolisiert. Zwischen den einzelnen Schritten wurden lange Pausen eingebaut, eingezeichnet sind nur die Pausen während des Bildladevorganges.

Die Messungen wurden auf die fünf zuvor verwendeten CR-Bilder beschränkt, da 33 Phototransistoren nötig gewesen wären, um Messungen mit 4 x 4 Tableaus von CT oder MR-Bildern durchzuführen. Jedes dieser kleinen Bilder müßte einzeln überwacht werden und jedes bringt sein eigenes Fortschrittsicon mit, das auch hätte überwacht werden müssen. Die Ansteuerung dieser Zahl

von Phototransistoren hätte wiederum eine wesentlich komplexere externe Beschaltung notwendig gemacht.

Zwischen den verschiedenen Durchgängen wurde das BVS-Klient-Applet und der Internet Explorer beendet und neu gestartet, um die Bilder aus dem lokalen Speicher zu entfernen.

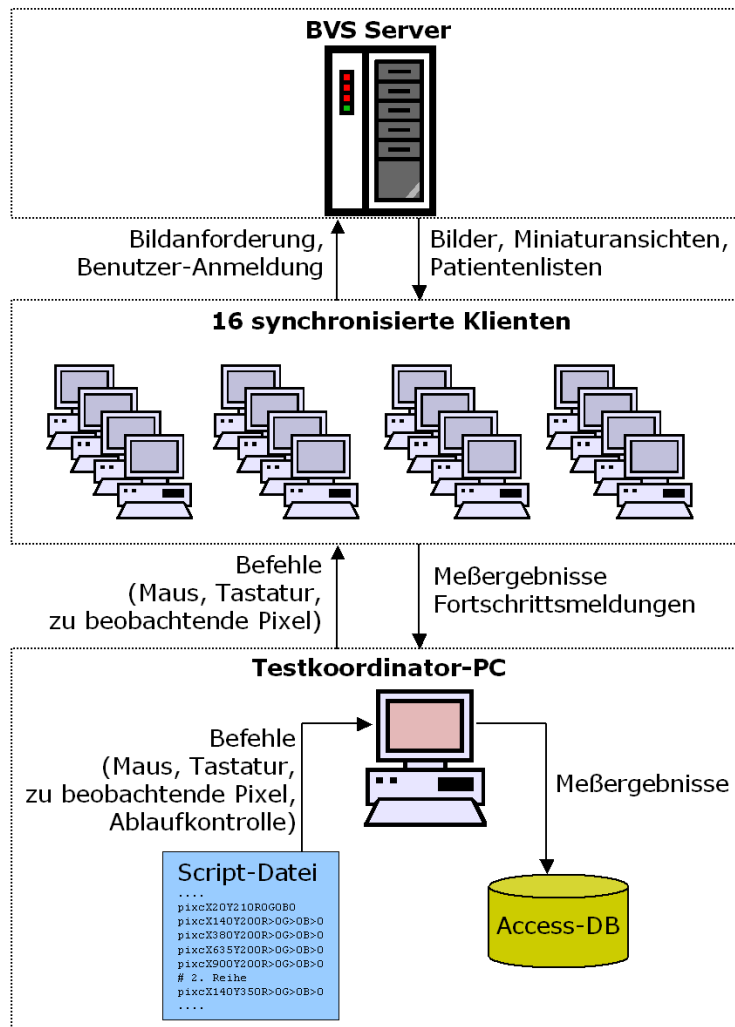
### **3.6.4 Methode 3: Auslesen der Farbe von Bildschirmpunkten**

Im Anschluß wurde eine weitere Methode entwickelt, die auf dem direkten Auslesen von Bildschirmpunkten (Pixeln) innerhalb des zu beobachtenden PC basierte. Zu diesem Zweck wurde eine Anwendung in MS Visual Basic 6.0 SP4 (Microsoft® (MS), Redmond, WA, U.S.A) geschrieben, die mit Hilfe von Application Programming Interface (API)-Aufrufen an die gdi32.dll die genaue Farbe bestimmter Pixel auf dem Bildschirm auslesen konnte. Das API und die gdi32.dll bilden einen Teil des Windows Betriebssystems. Darüber hinaus war die entwickelte Anwendung in der Lage, Mausklicks auf bestimmte Bildschirmregionen und Tastatureingaben zu simulieren. Ähnlich wie bei der Anwendung von Phototransistoren wurden bestimmte Pixel gewählt, aus deren Farbe der Fortschritt eines Ladevorganges abgelesen werden konnte. Die Meßergebnisse wurden anschließend in einer MS Access 2000 (Microsoft® (MS), Redmond, WA, U.S.A) Datenbank gespeichert.

Die erste Version dieser Applikation verwendete eine einfache lokal gespeicherte Textdatei, um Anweisungen für die zu beobachtenden Pixel und die auszuführenden Mausklicks zu speichern. Aus den in Textdateien gespeicherten Befehlsketten ergaben sich Scripts, die die für bestimmte Testvorgänge nötigen Kommandofolgen enthielten.

Im Rahmen der späteren Untersuchungen mit mehreren Klient-Computern wurde dieses Programm in zwei Komponenten aufgeteilt, die auf verschiedenen Rechnern liefen und Daten über das Netzwerk austauschten. Die Server-Komponente, die auf einem separaten als Testkoordinator-PC bezeichneten Rechner ausgeführt wurde, las die Textdateien mit den Instruktionen und übermittelte diese gleichzeitig an eine beliebige Zahl verbundener Klienten. Die Klient-Komponente führte die übermittelten Aktionen aus und sendete die Meßergebnisse zurück an den Testkoordinator-PC, der sie in einer

MS Access 2000 Datenbank speicherte. Durch dieses Vorgehen konnten viele Rechner synchron Aktionen durchführen und so gegenüber dem BVS große Last simulieren.



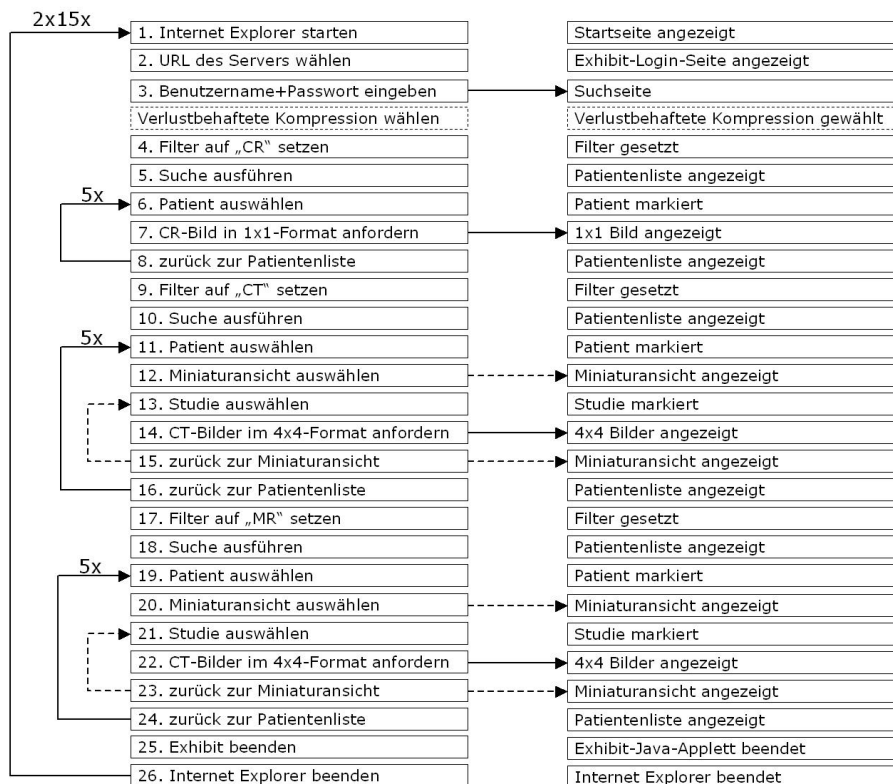
**Abbildung 13: Testaufbau während der Tests mit Methode 3 und 16 synchronisierten Klienten**

Während der Klient-Hardware-Tests wurde jeweils nur ein Klient betrieben, im Gegensatz dazu wurden während der Tests bei gleichzeitigem Hochladen und während der Lasttests bis zu 16 Klienten synchronisiert betrieben.

Die vollständige Automatisierung der Tests wurde durch Scripts erreicht, die die Abfolge von Vorgängen auf den einzelnen Klienten kontrollierten. Die Erstellung dieser Scripts umfaßte die Auswahl von Pixeln, auf die geklickt werden sollte und die Auswahl solcher, die während des Herunterladens bestimmte Farben erreichen sollten.

Alle Scripts basierten auf dem gleichen Ablauf (Abbildung 14, Seite 37). Durch eine Reihe simulierter Mausoperationen und Tastatureingaben wurde der Internet Explorer gestartet und die BVS-Startseite aufgerufen. Im Anschluß wurde das BVS-Klient-Applet geladen und eine Anmeldung am Server ausgeführt. Danach wurde eine Studie bzw. ein Bild gewählt. Nun wurden alle im Rahmen des Herunterladens zu beobachtenden Pixel und deren erwartete Farbwerte an den Klient übertragen und dort gespeichert, um Netzwerkverkehr während des eigentlichen Download zu vermeiden. Durch einen simulierten Klick wurde jetzt der Ladevorgang und zugleich die Stoppuhr gestartet. Nun wurde im 0,01-Sekundentakt der erste Pixel beobachtet, der so lokalisiert war, daß er nach Umschalten in den Bildanzeigemodus des BVS-Klient-Applets schwarz wurde. Sobald dieser die erwartete Farbe erreichte, wurde die Überwachung auf den nächsten Pixel geschaltet, der sich im eigentlichen Bild befand. Erreichte dieser die Zielfarbe, so wurde der nächste beobachtet und so weiter. Der letzte beobachtete Pixel befand sich auf dem Icon des BVS-Klient-Applets, das einen Ladevorgang signalisiert. Nachdem alle Pixel die erwarteten Farben erreicht hatten, wurde die Stoppuhr angehalten und der Meßwert an den Testkoordinator-PC übermittelt. Im Anschluß wurde durch simulierte Mausoperationen die nächste Studie ausgewählt und der Vorgang wiederholt. Zwischen den verschiedenen Durchgängen, jedoch nicht zwischen Bildtypen, wurde das BVS-Klient-Applet und der Internet Explorer beendet und neu gestartet, um die Bilder aus dem lokalen Speicher zu entfernen.

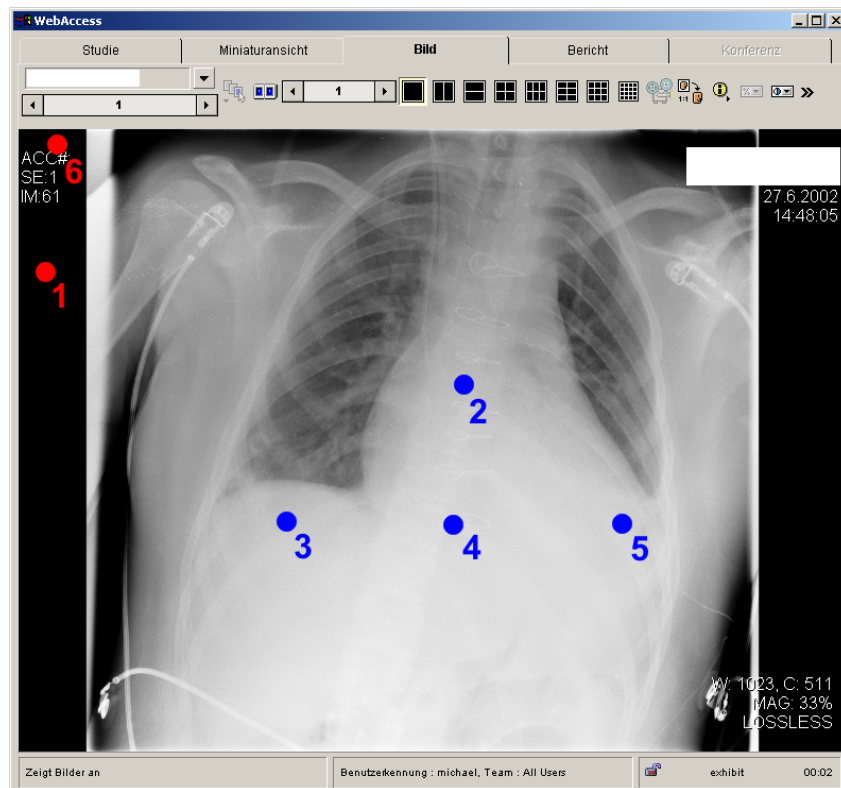
Um eine saubere Ausführung der Scripts zu gewährleisten, wurden auch außerhalb der eigentlichen Messungen immer wieder einzelne Pixel geprüft, um sicherzustellen, daß sich das BVS-Klient-Applet im erwarteten Zustand befand. War dies nicht der Fall, so wurde zunächst abgewartet und nach Ablauf einer vorgegebenen Zeit erneut geprüft. Bei Fortbestehen des Problems wurde der Testbetreuer akustisch unter Angabe des betroffenen Rechners auf das Problem hingewiesen.



**Abbildung 14: Schematischer Ablauf der Messungen mit Methode 3.** In der linken Spalte sind die Aktionen aufgeführt, rechts die jeweiligen Ergebnisse. Verbindungspfeile symbolisieren Zeitmessungen. Links sind die Wiederholungen durch Pfeile symbolisiert. Gestrichelte Aktionen werden je nach Status der Testsequenz durchgeführt. Längere Pausen waren nicht notwendig, da durch die Beobachtung von Bildschirmpunkten ein direkter Feedback zwischen BVS-Klient-Applet und der Meß-Applikation bestand.

Bei allen Bildtypen wurde nur die zur Sicherstellung des Abschlusses eines Ladevorganges minimal nötige Anzahl an Pixeln überwacht. Für CRs wurden sechs Pixel überwacht (Abbildung 15), bei CT- und MR-Serien waren es 33 Pixel (Abbildung 16). Bei allen Bildtypen lag der erste Pixel außerhalb des Bildes, um sicherzustellen, daß das BVS-Klient-Applet in den Anzeigemodus umgeschaltet hatte. Bei CR-Einzelbildern lagen drei Pixel unterhalb des Diaphragma und einer im Mediastinum. Diese erreichten bereits beim Eintreffen der ersten unscharfen Version des Bildes den erwarteten Grauton. Der letzte Pixel wurde so positioniert, daß er innerhalb des kleinen Icons lag, das als Teil des BVS-Klient-Applets den Fortschritt von Ladevorgängen anzeigt und nach Abschluß des Vorgangs verschwindet (Abbildung 5, Seite 24). Bei den aus 16 Bildern bestehenden Tableaus von CT- und MR-Bildern wurde innerhalb jedes Bildes ein Pixel beobachtet. Da jedes Bild sein eigenes Fortschrittsicon mitbringt, mußten 16 weitere Pixel innerhalb dieser Fortschrittsicons beobachtet

werden. Um die CPU und den Rechner zu entlasten, wurde jeweils nur ein Pixel beobachtet, bis dieser die gewünschte Farbe erreichte. Erst dann ging die Beobachtung auf den nächsten Pixel über.



**Abbildung 15: CR-Bild in Einzelansicht. Der Kompressionstyp ist rechts unten ein-geblendet (lossless). An den roten Punkten wurde auf „schwarz“ gewartet, an den blauen auf Werte im Graubereich. Die Nummern geben die Reihenfolge an, in der die Pixel beobachtet wurden.**

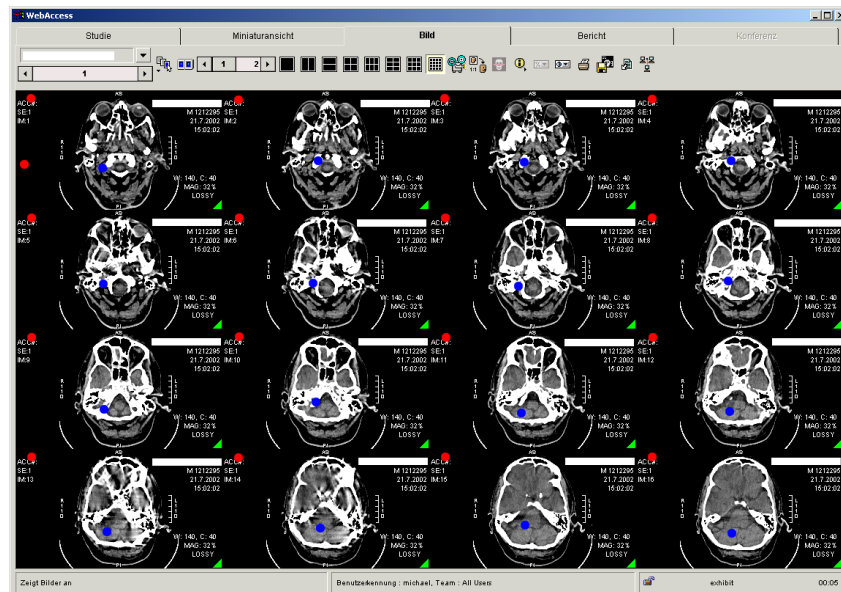


Abbildung 16: 16 CT-Bilder als 4 x 4-Tableau. Der gewählte Kompressionstyp ist in diesem Fall verlustbehaftet (lossy) und wird bei jedem Bild rechts unten angezeigt und zusätzlich durch das grüne Dreieck symbolisiert. Hier wurden 33 Pixel beobachtet. An den roten Punkten wurde auf „schwarz“ gewartet, an den blauen auf Werte im Graubereich.

Während der Tests mit mehreren Klienten wurde die Übermittlung von Aufträgen durch den Testkoordinator-PC erst dann fortgesetzt, wenn der letzte Klient sein Meßergebnis übermittelt hatte. Dadurch mußten alle Klienten auf den langsamsten warten, der nächste Ladevorgang fand dann wieder synchron statt.

Die einfache Scriptsprache, die die Mauseaktionen und die zu beobachtenden Pixel definierte, umfaßte Befehle wie leftX100Y200, um einen Mausklick mit der linken Maustaste auf Pixel 100, 200 auszuführen oder moveX500Y123, um den Mauszeiger auf dem Bildschirm zu bewegen. Die nach einer Messung erwarteten Bildschirmfarben wurden über Befehle wie pixcX527Y450R150G150B150 definiert, die wie in diesem Fall eine bestimmte Farbe des Pixels 527, 450 erwarteten. Pixelfarben konnten auch als Bereiche angegeben werden.

Ein Meßvorgang für ein CR-Einzelbild setzte sich wie in Tabelle 5 angegeben zusammen:

# Ersten Patienten wählen  
leftX167Y254

Den ersten Patienten in der Liste durch

wait0000.5	Klicken mit der linken Maustaste auswählen.
ccolX784Y253R0G0B255	Eine halbe Sekunde warten.
	Prüfen, ob der erste Patient auch tatsächlich ausgewählt ist (blau hervorgehoben). Der Server wartet, bis alle Klienten Erfolg melden.
wait0001	Eine Sekunde warten.
strtDummyCR	Klienten informieren, daß ein Meßvorgang ansteht.
# Links neben Bild pixcX50Y338R0G0B0 # Im Mediastinum pixcX527Y450R>130G>130B>130 # 3x unterhalb des Zwerchfells pixcX400Y650R>130G>130B>130 pixcX500Y650R>130G>130B>130 pixcX600Y650R>130G>130B>130 # Links neben Bild pixcX50Y338R0G0B0 # Fortschrittsicon pixcX35Y125R0G0B0 wait0000.5	Übertragung der zu beobachtenden Pixel an alle am Meßvorgang beteiligten Klienten. Die zu beobachtenden Pixel werden von den Klienten lokal gespeichert, um Netzwerkverkehr während des Meßvorgangs zu minimieren.
leftX247Y82	0,5 Sekunden Pause zur Sicherheit.
	Klick mit der linken Maustaste aller Klienten auf das Anzeige-Icon. Der Bildladevorgang startet in diesem Moment. Gleichzeitig starten alle Klienten die Stopuhr und die Überwachung der zuvor übertragenen Pixel. Nachdem alle Pixel die angegebenen Farben erreicht haben, wird das Ergebnis an den Server übermittelt, der dieses zusammen mit der hinter strt angegebenen Bezeichnung in der Datenbank ablegt.
wall	Abwarten, bis alle Klienten Ergebnisse gemeldet haben.

**Tabelle 5: Abfolge der Script-Befehle eines Meßvorganges unter Verwendung von Methode 3.**



Zudem war es möglich, durch den Befehl `cres` die Bildschirmauflösung und Farbtiefe eines Klienten umzustellen, wobei `cres1600x1200x8` eine Auflösung von 1600 x 1200 Pixeln bei einer Farbtiefe von 8 Bit einstellte. Durch den `dblb`-Befehl konnte eine Notiz in der Ergebnis-Datenbank gespeichert werden, um den Testverlauf zu dokumentieren. Um bei fehlgeschlagenen Operationen ggf. weiterarbeiten zu können, wurde der Befehl `tout` eingeführt, der den Server anwies, nach `n` Sekunden einfach weiterzumachen, als sei die Operation erfolgreich.

In der Regel wurden alle Befehle immer über das Netzwerk vom Testkoordinator-PC an alle Klienten weitergegeben. Bestimmte Vorgänge sollten jedoch nur von einem bestimmten Klienten durchgeführt werden, beispielsweise im Rahmen der Tests mit verschiedenen Bildern oder bei der Anmeldung sollten bei allen Rechnern verschiedene Benutzernamen verwendet werden. Dies wurde durch den Befehl `specific` ermöglicht. Nach der Angabe `specific 2,5,7` wurden mit allen weiteren Befehlen nur die Rechner mit den Nummern 2, 5 und 7 angesprochen bis entweder ein neuer `specific`-Befehl folgte oder `endspecific` wieder alle Klienten freischaltete.

Die Synchronisierung mehrerer Klienten machte den Befehl `ccolX527Y450R150G150B150` notwendig, der ohne Zeitmessung sicherstellte, daß auf allen Rechnern der angegebene Pixel den angegebenen Farbwert erreicht hatte. Erst wenn alle Rechner dies bestätigt hatten, wurde die Befehlsreihe fortgesetzt.

Die Scriptsprache erlaubte darüber hinaus Wiederholungen (loops) und Subroutinen sowie das Einbinden externer Scriptdateien, um häufige Wiederholungen zu vereinfachen. Konditionale Ausführung und Variablen wurden nicht implementiert.

Zur Auswertung der Daten wurde für jede Methode eine Reihe kleiner Anwendungen entwickelt, die die Meßergebnisse einer Prüfung unterzogen und sie in MS Excel Arbeitsmappen (Microsoft® (MS), Redmond, WA, U.S.A) speicherten.

Im Laufe der verschiedenen Testläufe wurden allein mit Methode 3 über 500.000 Vorgänge gemessen.

## **3.7 Server Überwachungsanwendung**

Für die Untersuchungen während des Hochladens von Bildern wurde ein zusätzliches Tool entwickelt, das auf dem BVS-Server lief und dessen Leistungsindikatoren sowie die Anzahl von Dateien in bestimmten Verzeichnissen in vorgegebenen Abständen protokollierte. Die von diesem Tool beobachteten Parameter setzen sich aus drei Leistungsindikatoren zusammen, namentlich Netzwerkverkehr, Prozessornutzung des Storage Service Class Providers (S-SCP) und des Curator. Die übrigen beiden Parameter waren die Anzahl der Dateien in den Verzeichnissen „Image Cache“ und „Web Cache“.

## **3.8 Vortests**

### **3.8.1 Tests unserer Umgebung**

Vor Beginn der Untersuchungen wurden mehrere Vortests durchgeführt, um unseren Testaufbau zu verifizieren. Zunächst wurde untersucht, ob die Tatsache, daß die Tests im regulären Netzwerk der Uniklinik Frankfurt (KliNet) stattfinden, einen Einfluß auf die Ergebnisse haben könnte. Zu diesem Zweck wurden mit Methode 1 Bilder angefordert, während Klient und Server zum einen im KliNet, zum anderen an einem Exklusiven Cisco Catalyst 3500 (Cisco Technology, San Jose, CA, U.S.A.) angebunden waren.

Um zu prüfen, ob eine große Zahl nicht-synchronisierter Klienten einen Einfluß auf die Anzeigezeiten von Bildern hat, wurde auch mit Methode 1 ein Test mit acht nicht synchronisierten Rechnern durchgeführt und das Ergebnis mit einem einzelnen Klienten verglichen. Zu diesem Zweck wurden fünf CR- und je zwei CT- und MR-Serien 52 mal auf einen Testklienten übertragen. Dies wurde im Anschluß 54 mal wiederholt, während sieben weitere Testklienten mit Auto-mouse automatisiert aber unkoordiniert Bilder anforderten. Dabei kam es praktisch nie zu gleichzeitigen Anforderungen, da der Vorgang der Anforderung im Gegensatz zu den übrigen Aktivitäten wie Auswahl des Patienten oder Anmelden am Server nur einen sehr kleinen Zeitanteil ausmacht.

Schließlich wurde untersucht, ob das Laden gleicher oder verschiedener Bilder durch die einzelnen Klienten einen Einfluß auf die Anzeigezeit hat. Dieser Test

wurde mit Methode 3 durchgeführt und je fünfmal wiederholt. Dabei luden 8 bzw. 16 synchronisierte Klienten gleiche oder verschiedene Bilder herunter.

### **3.8.2 Untersuchung des BVS**

Im Rahmen der Messungen mit Methode 1 wurden mehrere Male Bilder im BVS-Klient-Applet angezeigt, während mit Hilfe von perfmon oder dem selbstentwickelten Tool Leistungsindikatoren aufgezeichnet wurden. Dies geschah, um die Prozesse im BVS-Klient-Applet besser kennen zu lernen.

Vor den Untersuchungen beim Hochladen von Daten wurden zwei Vortests durchgeführt. Ausgehend von den oben beschriebenen Sender-PCs wurden verschieden große Bildserien an den Server übertragen. Zunächst wurde die Server-Überwachungsanwendung während des Uploads einer aus 590 Bildern bestehenden CT-Studie ausgeführt, anschließend wurden fünf separate CT-Studien mit zusammen 791 Bildern (je 43, 125, 244, 103, 187 und 89 Bilder) hochgeladen und die Meßwerte protokolliert. Die Ergebnisse dieses Vortests waren für das Verständnis der weiteren Tests mit dem Hochladen von Bildern notwendig.

Im Rahmen des zweiten Vortests wurde der gesamte Inhalt der drei Sender-PCs übertragen. Zunächst erfolgte dies für jeden Sender-PC einzeln. Nach Löschen aller Bilder auf dem Server wurde der Vorgang wiederholt, indem alle drei Sender-PC gleichzeitig Bilder schickten.

## **3.9 Testserien**

### **3.9.1 Klient-Hardware-Modifikationen**

Im Rahmen der Klient-Messungen wurden verschiedene Konfigurationen und Leistungsklassen von Klient-PCs getestet und verglichen.

Alle Tests wurden mit Hilfe der oben beschriebenen Methode 3 durchgeführt. Zeitmessungen wurden für das Login, 16 Bildladevorgänge (Dummy, 5 verschiedene CR, 5 verschiedene CT und 5 verschiedene MR), und 10 Miniaturansichten (5 CT und 5 MR) definiert [5, 30]. Der Meßvorgang wurde für jede Konfiguration 15 mal mit verlustfreier und 15 mal mit verlustbehafteter

Kompression wiederholt. Ein kompletter Testdurchlauf beinhaltete 1.050 Zeitmessungen, von denen 450 Bildmessungen waren.

Die Ergebnisse wurden pro Bildtyp und Kompressionsverfahren gemittelt, so daß sich folgende sechs Kennzahlen ergaben: Durchschnittliche Anzeigezeit von CR, CT und MR jeweils bei verlustfreier und verlustbehafteter Kompression.

Es wurden sieben Test-PCs eingesetzt, die jeweils in einer Standardkonfiguration zur Verfügung standen (Tabelle 6). Diese Standardkonfigurationen unterschieden sich nur durch die Prozessorgeneration und Taktfrequenz. Aus jeder Prozessorgeneration wurde ein Referenz-PC ausgewählt (2, 4, 6), an dem verschiedene Modifikationen vorgenommen wurden. Die Modifikationen umfaßten Änderungen an der RAM-Ausstattung sowie verschiedene Netzwerk und Grafikeinstellungen. Eine Übersicht über die Grafikeinstellungen wurde in Tabelle 7 zusammengestellt. Die verbleibenden PCs 1, 3, 5 und 7 wurden nur in dieser Konfiguration mit einer Peripheral Component Interconnect (PCI)-Grafikkarte mit 4 MB RAM, einer Bildschirmauflösung von 1024 x 768 Pixeln und einer Farbtiefe von 16 Bit gemessen.

PC Nummer	1		2		3		4		5		6		7	
Prozessortyp, Takt (MHz)	PI, 90		PI, 166		PI, 233		PII, 350		PIII, 500		PIII, 700		PIII, 933	
Konfiguration	Stand.		Stand. Modi.		Stand.		Stand. Modi.		Stand.		Stand. Modi.		Stand.	
RAM (MB)	128		128 64		128		128 64 256		128		128 64 256		128	
Netzwerkanbindung (Mbit/s)	100F		100F 10H		100F		100F 10H		100F		100F 10H		100F	
(F=Voll duplex, H=Halbduplex)			10F 100H				10F 100H				10F 100H			

**Tabelle 6: Hardwarekonfigurationen der getesteten Klient-PCs. Alle sieben PCs standen in einer Standardkonfiguration (Stand.) zur Verfügung, an der nur Prozessortyp und Taktfrequenz geändert wurden. Die PCs 1, 3, 5, 7 wurden nur in dieser Standardkonfiguration mit einer 4 MB PCI Grafikkarte und einer Bildschirmauflösung von 1024 x 768 Pixeln bei einer Farbtiefe von 16 Bit getestet. Aus jeder Prozessorgeneration wurde ein Referenz-PC gewählt (2, 4, 6), an dem weitere Modifikationen vorgenommen wurden (Modi.). RAM und Netzwerkeinstellungen sind dieser Tabelle zu entnehmen, Bildschirmauflösungen gehen aus Tabelle 7 hervor. Es wurde jeweils nur eine Modifikation vorgenommen, alle anderen blieben in Standardeinstellung. *PI* Pentium I; *PII* Pentium II; *PIII* Pentium III.**

PC Nummer	2 (PI 166)			4 (PII 350)				6 (PIII 700)																			
Grafikkarte																											
RAM (MB), Bustyp	2, PCI		4, PCI	2, PCI		4, PCI	4, AGP		8, AGP	2, PCI		4, PCI	4, AGP		8, AGP												
Farbtiefe (Bit)	8	16	24	8	16	24	8	16	24	8	16	24	8	16	24												
Auflösung (Pixel)																											
1024x768	+	+	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+			
1152x864	+	+	-	+	+	+	+	-	-	+	+	+	+	+	+	+	-	-	+	+	+	+	+	+	+	+	+
1280x1024	+	-	-	+	+	+	+	-	-	+	+	-	+	+	-	+	+	-	+	+	-	+	+	-	+	+	+
1600x1200	-	-	-	+	+	-	-	-	-	+	+	-	+	+	-	+	+	-	-	-	-	+	+	-	+	+	+

**Tabelle 7: Zusätzlich zu den Konfigurationen aus Tabelle 6 wurden die drei Standard-PCs mit bis zu vier verschiedenen Grafikkarten ausgestattet. Mit jeder Grafikkarte wurden drei verschiedene Farbtiefen (8, 16 und 32 Bit) und vier Bildschirmauflösungen getestet. Mit „-“ markierte Kombinationen waren aus technischen Gründen nicht möglich. Wie bei Tabelle 6 wurde jeweils nur ein Parameter geändert. PCI: Peripheral Component Interconnect; AGP Accelerated Graphics Port; PI Pentium I; PII Pentium II; PIII Pentium III.**

Insgesamt wurden Messungen mit 113 verschiedenen Klient-Konfigurationen durchgeführt. Um den Einfluß der einzelnen Komponenten möglichst genau eingrenzen zu können, wurde jeweils nur ein Parameter geändert. 21 Konfigurationen ergaben sich aus Änderungen an Prozessorgeschwindigkeit, RAM oder Netzwerkeinstellungen (10 bzw. 100 Mbit/s und Voll- gegen Halbduplex) (Tabelle 6). Weitere 86 Konfigurationen entstanden durch Änderung der Bildschirmauflösung und Farbtiefe (Tabelle 7). Diese 107 Konfigurationen wurden unter Verwendung des Betriebssystems Windows NT 4.0 SP 6 getestet. Weitere 6 Tests wurden mit Windows 2000 durchgeführt. Für jede Konfiguration wurde ein kompletter Testdurchlauf durchgeführt, so daß am Ende 118.500 Zeitmessungen durchgeführt wurden, von denen 50.850 Bildanzeigeweiten waren. Unter Berücksichtigung der 4 x 4-Tableaus ergeben sich 559.350 heruntergeladene Bilder.

### 3.9.2 Hochlade-Tests

Im Rahmen der Hochlade-Tests sollte das Verhalten des Servers bei gleichzeitigem Hochladen von Bildern ermittelt werden. Um ausreichend Platz zu schaffen, wurde das vorhandene RAID-V in ein RAID-0 (Stripeset ohne Parity) umgewandelt. Bei dieser Untersuchung wurde der oben beschriebene Standardserver durch Entfernen von 768 MB RAM und eines Prozessors zu einem „Kleinen Server“ (ein Prozessor, 256 MB RAM) reduziert und die Ergebnisse mit denen des Standardservers verglichen.

### 3.9.2.1 Hochladekapazität

Für jeden Testdurchlauf wurde der vollständige Inhalt eines bestimmten Sender-PCs übertragen und die Dauer dieses Vorganges sowie der Nachbearbeitung gemessen. Dies wurde fünfmal ohne gleichzeitig stattfindendes Herunterladen und Anzeigen wiederholt. Anschließend wurden diese fünf Durchgänge wiederholt, während 16 Klienten gleichzeitig immer wieder Bilder abholten und anzeigten. Die Ergebnisse jedes Durchlaufes wurden gemittelt.

Dies wurde für drei verschiedene Serverkonfigurationen durchgeführt, zum einen mit dem Standardserver mit verlustfreier Kompression, zum zweiten mit dem „Kleinen Server“ mit verlustfreier Primärkompression und zum dritten mit dem „Kleinen Server“ mit verlustbehafteter Primärkompression.

Daraus ergaben sich 18 Kennzahlen, nämlich für jede der drei Serverkonfigurationen drei verschiedene Bildtypen jeweils mit und ohne gleichzeitiges Herunterladen. Aus diesen Zahlen ließ sich bei bekannter hochgeladener Datenmenge die Hochladekapazität in MB pro Stunde für jede Konfiguration und Situation errechnen.

### 3.9.2.2 Herunterladegeschwindigkeit

Wie bereits erwähnt, wurden diese Messungen mit anderen Bildern als denen, die für die Server- und Klient-Messungen verwendet wurden, durchgeführt. Dies war nötig, um sicherzustellen, daß die während der Tests automatisch ausgewählten Patienten bei zunehmender Zahl von Bildern auf dem Server nicht ihre Position in der Auswahlliste änderten und somit den Ablauf der Scripts, die auf vordefinierte Bildschirmregionen klickten, störten. Die Namen der entsprechenden neuen Patienten begannen mit A und B, so daß diese immer an oberster Stelle in der Auswahlliste verblieben. Die Charakteristika der Bilder entsprachen denen der für die anderen Messungen verwendeten.

Um vergleichbare Daten zu erhalten, wurden zunächst die Messungen mit 16 gleichzeitig Bilder anfordernden Klienten durchgeführt, ohne daß Bilder hochgeladen wurden. Im Anschluß wurden diese Messungen wiederholt, während die verschiedenen Bildtypen von den einzelnen Sender-PCs an das BVS übertragen wurden. Dies wurde für alle drei Servertypen, namentlich

Standardserver, „Kleiner Server“ verlustfrei und „Kleiner Server“ verlustbehaftet durchgeführt.

Daraus ergaben sich neun Kombinationen aus drei Servern und drei hochgeladenen Bildtypen. Mit diesen neun Kombinationen wurde die Herunterladezeit von allen drei verschiedenen Bildtypen jeweils verlustbehaftet oder verlustfrei gemessen. Daraus ergeben sich 9 Kombinationen \* 3 Bildtypen \* 2 Kompressionsverfahren = 54 Meßkombinationen. Da ein verlustfreies Herunterladen bei verlustbehafteter Primärkompression nicht möglich ist, verbleiben 45 Meßkombinationen. Für jede dieser Meßkombinationen wurden Bildanzeigezeiten für die drei Phasen des Hochladens bestimmt: Vor dem Hochladen, während der Übertragung und während der Nachbearbeitung. Daraus ergaben sich 135 Kennzahlen.

Im Anschluß wurde für den Standardserver bestimmt, welchen Einfluß verschiedene Klientzahlen auf den Prozeß haben, indem alle 18 Kombinationen von drei hochgeladenen Bildtypen und drei jeweils verlustfrei und verlustbehaftet heruntergeladenen Bildtypen für 1, 2, 4, 8, 12 und 16 gleichzeitig anfordernden Klienten gemessen wurden.

### **3.9.3 Server-Modifikationen**

Für die Tests mit verschiedenen Server-Konfigurationen wurde Methode 3, so wie sie im Rahmen der Klient-Hardware-Tests durchgeführt wurde [11], angewandt und es wurden die gleichen Bilder wie bei diesen Tests verwendet. Bildanzeigezeiten wurden für fünf CR-, CT- und MR-Studien in verlustfreier und verlustbehafteter Kompression gemessen und dies fünfmal wiederholt. Im Rahmen eines Testlaufes wurden demnach 150 Bildanzeigezeiten bestimmt und unter Berücksichtigung der 4 x 4-Tableaus 1.650 Bilder heruntergeladen und angezeigt. Die Ergebnisse wurden pro Bildtyp und Kompressionsverfahren gemittelt.

Insgesamt wurden sechs verschiedene Serverkonfigurationen untersucht, wobei die auf Seite 22 beschriebene Standardkonfiguration jeweils um einen Parameter variiert wurde. Zunächst wurde der Arbeitsspeicher von 1 GB auf 512 MB und 256 MB reduziert. Nach Wiederherstellung des RAM wurde einer der beiden Prozessoren entfernt. Abschließend wurde der Standardserver

anstatt mit der FastEtherChannel-Anbindung mit 100 Mbit/s und 10 Mbit/s angebunden, letzteres mit nur einer Netzwerkkarte.

Jede dieser sechs Konfigurationen wurde mit einer steigenden Anzahl an Klient-PCs getestet, die durch die Software koordiniert gleichzeitig Bilder anforderten. Zunächst wurde die Standardkonfiguration mit einem bis 16 Klienten in Einerschritten getestet. Alle anderen Konfigurationen wurden mit neun verschiedenen Klientzahlen (1, 2, 4, 6, 8, 10, 12, 14 und 16) untersucht. Daraus ergaben sich 61 verschiedene Kombinationen aus Server-Konfiguration und Klientzahl, für die jeweils drei Bildtypen (CR, CT, MR) bei verlustbehafteter und verlustfreier Sekundärkompression betrachtet wurden, woraus 366 Kennzahlen ermittelt wurden. Für den Standardserver wurden 2.400 Bildanzeigezeiten bestimmt, für alle anderen Konfigurationen wurden insgesamt 6.750 Bildanzeigezeiten ermittelt. Unter Berücksichtigung der Tableaus ergeben sich daraus 74.250 heruntergeladene und angezeigte Bilder.

Zusätzlich dazu wurde der Testablauf so modifiziert, daß 8 bzw. 16 synchronisierte Klienten nicht zusammen das gleiche Bild anforderten, sondern verschiedene. Dies wurde nur mit der Standardkonfiguration durchgeführt und fünfmal wiederholt.



## 4 Ergebnisse

### 4.1 Entwicklung der Meßmethodik

#### 4.1.1 Qualität der Ergebnisse

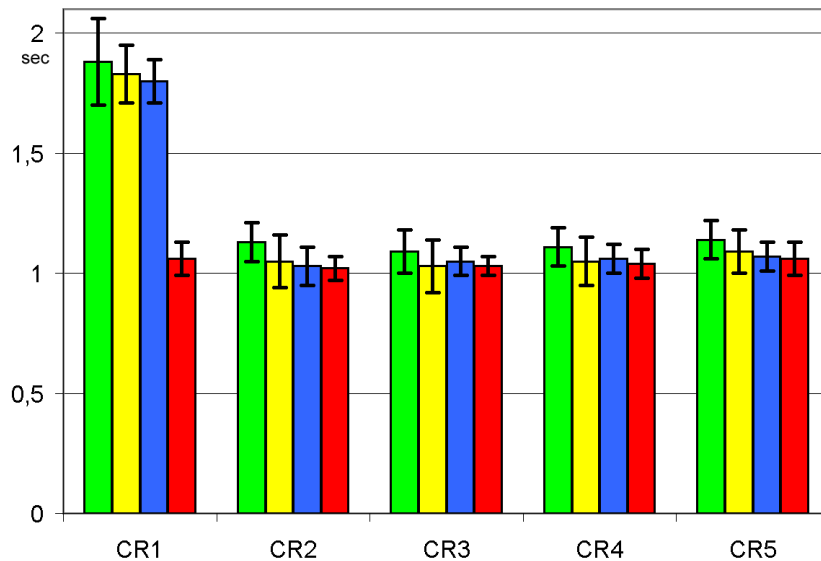
Tabelle 8 zeigt die Ergebnisse, die unter Anwendung der drei Methoden erzielt wurden. Bei allen Bildformaten produziert Methode 3 die geringste Standardabweichung, diese liegt je nach Bildtyp um Faktor 2 bis 3 niedriger als bei Methode 1. Wie erwähnt wurden aus technischen Gründen mit Methode 2 keine Tests mit CT- und MR-Tableaus durchgeführt.

	CR1	CR2	CR3	CR4	CR5	CT1	CT2	MR1	MR2
<b>Methode 1</b>									
Durchschnitt	1,88	1,13	1,09	1,11	1,14	3,75	3,20	2,67	2,32
Standardabw.	0,18	0,08	0,09	0,08	0,08	0,16	0,17	0,21	0,15
<b>Methode 1b</b>									
Durchschnitt	1,83	1,05	1,03	1,05	1,09	3,87	3,37	2,85	2,42
Standardabw.	0,12	0,11	0,11	0,10	0,09	0,16	0,09	0,16	0,11
<b>Methode 2</b>									
Durchschnitt	1,80	1,03	1,05	1,06	1,07				
Standardabw.	0,09	0,08	0,06	0,06	0,06				
<b>Methode 3</b>									
Durchschnitt	* 1,06	1,02	1,03	1,04	1,06	* 2,84	3,11	* 1,68	2,05
Standardabw.	0,07	0,05	0,04	0,06	0,07	0,04	0,07	0,03	0,04

**Tabelle 8: Anzeigezeiten für verschiedene Bildtypen mit verschiedenen Verfahren gemessen. \* Hinweis: Spätere Tests zeigten, daß der erste Anzeigevorgang nach dem Start des MSIE länger dauert als folgende. Aus diesem Grund lassen sich die Ergebnisse für CR1, CT1 und MR1 nicht zwischen Methode 3 einerseits und Methoden 1, 1b und 2 andererseits vergleichen.**

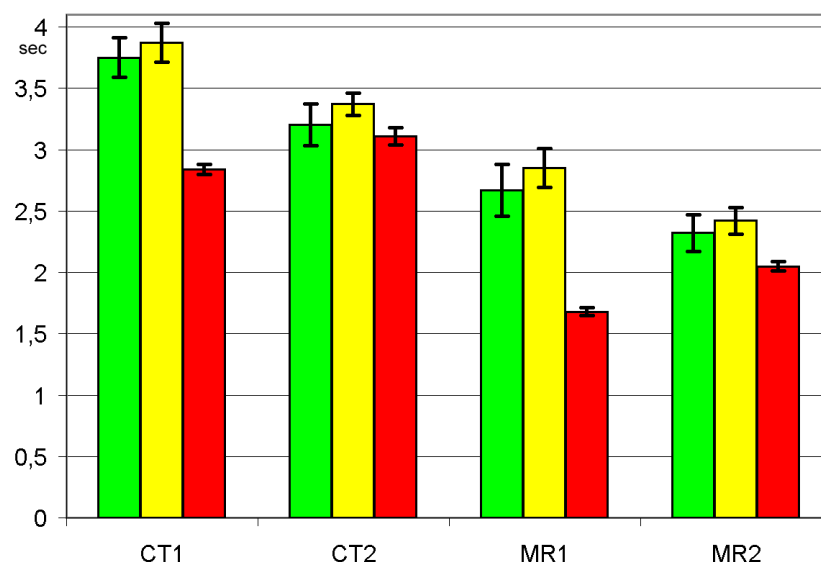
Allgemein gilt, daß alle drei Methoden beim Herunterladen von einzelnen CR-Bildern ähnliche Resultate lieferten. Die Standardabweichung lag jeweils unter 0,1, bei Methode 3 sogar um 0,06. Bei CR-Bildern lieferte Methode 1 4 % bis 10 % langsamere Meßwerte als Methode 2, während Methode 3 0 % bis 3 % schnellere Ergebnisse lieferte als Methode 2.

Abbildung 17 und Abbildung 18 zeigen die Ergebnisse und die Standardabweichung für die drei Methoden.



**Abbildung 17: Standardabweichungen und Absolutwerte der drei Methoden bei CR-Bildern. Aus technischen Gründen können die Ergebnisse für CR1, CT1 und MR1 nicht zwischen den Methoden 1, 1b und 2 verglichen werden. Grün: Methode 1; Gelb: Methode 1b; Blau: Methode 2; Rot: Methode 3.**

Methode 1b, die bei CR etwas schnellere Ergebnisse lieferte, produzierte bei CT- und MR-Tableaus deutlich langsamere Ergebnisse. Bei allen Bildtypen ergeben sich bei Messung mit Methode 3 die kürzesten Lade- und Anzeigeweiten.



**Abbildung 18: Standardabweichungen und Absolutwerte der drei Methoden bei CT- und MR-Serien. Aus technischen Gründen können die Ergebnisse für CR1, CT1 und MR1 nicht zwischen den Methoden 1, 1b und 2 verglichen werden. Grün: Methode 1; Gelb: Methode 1b; Blau: Methode 2; Rot: Methode 3.**

Die minimale Meßgenauigkeit der Methode 2 wurde durch die Wiederhol-  
frequenz des Bildschirms definiert, die bei 72 Hz lag. Der Aufbau mit den  
Phototransistoren war in der Lage, 650 Meßwerte pro Sekunde zu erfassen. Es  
zeigte sich dabei, daß die Phototransistoren auch bei für das menschliche Auge  
hell erscheinenden Arealen viele negative Werte lieferten, das Nachleuchten  
des Monitors hatte also keine signifikante Auswirkung.

Es ist wichtig darauf hinzuweisen, daß im Rahmen späterer Test heraus-  
gefunden wurde, daß der jeweils erste Bildladevorgang nach einem Start des  
BVS-Klient-Applets deutlich länger dauerte als spätere Ladevorgänge. Dieses  
Verhalten erklärt sich aus Vorgängen innerhalb des BVS. Diese Beobachtung  
muß bei der Bewertung der vorliegenden Ergebnisse beachtet werden.  
Während das BVS-Klient-Applet bei Methoden 1, 1b und 2 nach jedem Bildtyp  
beendet und neu gestartet wurde, geschah dies bei Methode 3 erst nach dem  
Herunterladen aller drei Bildtypen. Aus diesem Grund lassen sich die Ergeb-  
nisse für CR1, CT1 und MR1 nicht zwischen den Methoden 1, 1b, 2 und  
Methode 3 vergleichen, sondern nur innerhalb der Methoden 1, 1b und 2.

#### **4.1.2 Gebrauchsfähigkeit und Stabilität**

Alle drei Methoden waren auf menschliche Unterstützung angewiesen, sobald  
es innerhalb des BVS zu Fehlern kam und Fehlermeldungen angezeigt wurden.  
Fehlermeldungen mußten jeweils durch einen Klick auf „OK“ manuell bestätigt  
werden, andere Fehler erforderten entsprechende Maßnahmen, um behoben  
zu werden. Diese Maßnahmen waren nicht automatisierbar.

Ein wesentliches Problem bei Methode 1 lag darin, daß die gemessenen  
Phasen mit hoher CPU-Aktivität nicht sicher bestimmten Vorgängen innerhalb  
des Bildverteilungssystems zugeordnet werden konnten. Dies konnte nur zum  
Teil durch automatische Sortierung anhand von Regeln übernommen werden,  
eine manuelle Prüfung jedes einzelnen Wertes war dennoch nötig. Zudem kam  
es vor, daß sehr langsame Ladevorgänge keine hohe CPU-Belastung zur Folge  
hatten, da die Daten langsam ankamen und von der CPU ohne besondere  
Anstrengung bewältigt werden konnten.

Die Zuordnung von Meßwerten zu Vorgängen war bei Methode 2 unpro-  
blematisch, da Ladevorgänge vom externen PC aus gestartet wurden und

dieser jederzeit wußte, welcher Vorgang anstand. Auch langsame Bildübertragungen führten zu keinen falschen Ergebnissen, da das Icon des BVS-Klient-Applets, das einen Ladevorgang signalisiert, erst bei vollständigem Vorliegen der Daten verschwindet und somit die Messung erst dann beendet wurde. Nachteilig bei Methode 2 ist, daß die zu beobachtende Zahl der Bildschirmregionen durch die Zahl der Phototransistoren limitiert ist und daß die gemessenen Bereiche deutliche Helligkeitsunterschiede aufweisen müssen, damit diese wahrgenommen werden können.

Nach dem relativ hohen Aufwand der Programm- und Scriptentwicklung lief Methode 3 sehr stabil. Da das System sofort nach dem Ende einer Messung mit der Vorbereitung der nächsten Messung beginnen konnte, wurden mit Methode 3 in der gleichen Zeit mehr Meßwerte erfaßt als mit Methode 1 und 2. Der Ablauf blieb nur dann stehen, wenn das BVS Fehler produzierte und Fehlermeldungen angezeigt wurden bzw. Bilder nicht korrekt geladen wurden. Die Auswahl der zu beobachtenden Pixel stellte bei CR-Bildern kein Problem dar, bei CT- und MR-Bildern hingegen mußte darauf geachtet werden, daß der beobachtete Pixel nicht in einem Bereich des Bildes lag, der schwarz blieb. Gerade bei craniellen CTs und MRs bleiben große Bereiche auch im Zentrum des Bildes dunkel. Lag der beobachtete Pixel in einem dieser Bereiche, konnte das Eintreffen des Bildes nicht festgestellt werden. Dies führte zu erheblichem Aufwand beim Erzeugen der Scripts, da auf jedem Tableau andere Pixel beobachtet werden mußten. Zudem mußten die Scripts natürlich auf jede Bildschirmauflösung angepaßt werden. Die Möglichkeit, Subroutinen innerhalb der Scripts zu definieren und externe Dateien einzubinden, erlaubte jedoch eine recht übersichtliche Gestaltung der Scripts und Wiederverwendung großer Scriptteile.

## **4.2 Vortests**

### **4.2.1 Tests unserer Umgebung**

Es zeigt sich, daß zwischen dem KliNet und einem exklusiven Netz kein signifikanter Unterschied besteht. Die Differenzen liegen in allen Fällen unter 1 % der Meßwerte und zeigen keine Richtung.

Mittelwert	CR1	CR2	CR3	CR4	CR5	CT1	CT2	MR1	MR2
KliNet	1,88	1,13	1,09	1,11	1,14	3,75	3,20	2,67	2,32
Exklusiv	1,84	1,10	1,11	1,13	1,13	3,78	3,23	2,74	2,33
Delta	0,04	0,02	-0,01	-0,01	0,01	-0,03	-0,02	-0,08	-0,01

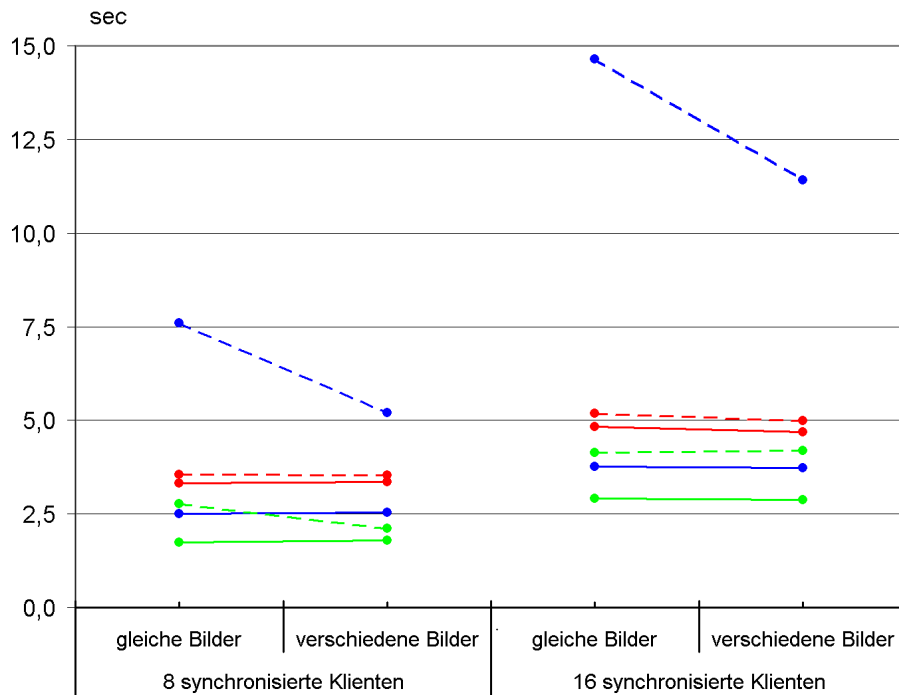
**Tabelle 9: Mittelwerte unter Verwendung von Methode 1 in einem exklusiven Netzwerk und im KliNet.**

Es zeigt sich, daß der Server von mehreren nicht synchronisierten Klienten kaum merklich beeinflusst wird. Bei der Beobachtung fiel auf, daß nur in sehr seltenen Fällen tatsächlich gleichzeitige Übertragung an mehrere Klienten stattfand.

Delta	CR1	CR2	CR3	CR4	CR5	CT1	CT2	MR1	MR2
MW	0,086	-0,017	0,054	0,025	0,059	-0,021	0,002	-0,027	-0,017
MAX	0,020	0,090	0,182	0,130	0,128	-0,320	-0,145	-0,035	0,070
MIN	0,051	0,061	0,122	0,092	0,110	0,000	-0,100	-0,560	-0,090
STABW	0,001	0,015	0,004	0,002	0,003	-0,034	-0,039	0,045	0,009

**Tabelle 10: Ergebnisse mit nicht synchronisierten Klienten unter Verwendung von Methode 1.**

Der zusätzliche Test (Abbildung 19) zeigte, daß es bei verlustfreier Sekundärkompression keinen Einfluß hatte, ob die Klienten das gleiche oder verschiedene Bilder anforderten. Allerdings ergab sich bei verlustbehafteter Sekundärkompression ein klarer Abfall der Anzeigezeit als verschiedene MR-Bilder geladen wurden. Bei CT-Bildern konnte praktisch kein Unterschied festgestellt werden.



**Abbildung 19: Anzeigezeiten (in Sekunden) abhängig davon, ob von den 8 oder 16 synchronisierten Klienten gleiche oder verschiedene Bilder angefordert wurden. Grün: CR; Blau: MR; Rot: CT; Durchgezogene Linie: verlustfreie Kompression, gestrichelte Linie: verlustbehaftete Kompression.**

#### 4.2.2 Untersuchung des BVS

Bei der Auswertung der Leistungsindikatoren auf den Klient-PCs fiel auf, daß das BVS-Klient-Applet bereits relativ lange vor der effektiven Netzwerkübertragung des Bildes bzw. der ersten Serie Aktivität aufwies. Bei späteren Bildern war dies nicht mehr der Fall. Es zeigt sich außerdem, daß nach der Übertragung über das Netzwerk über einen gewissen Zeitraum noch Aktivität verzeichnet wurde.

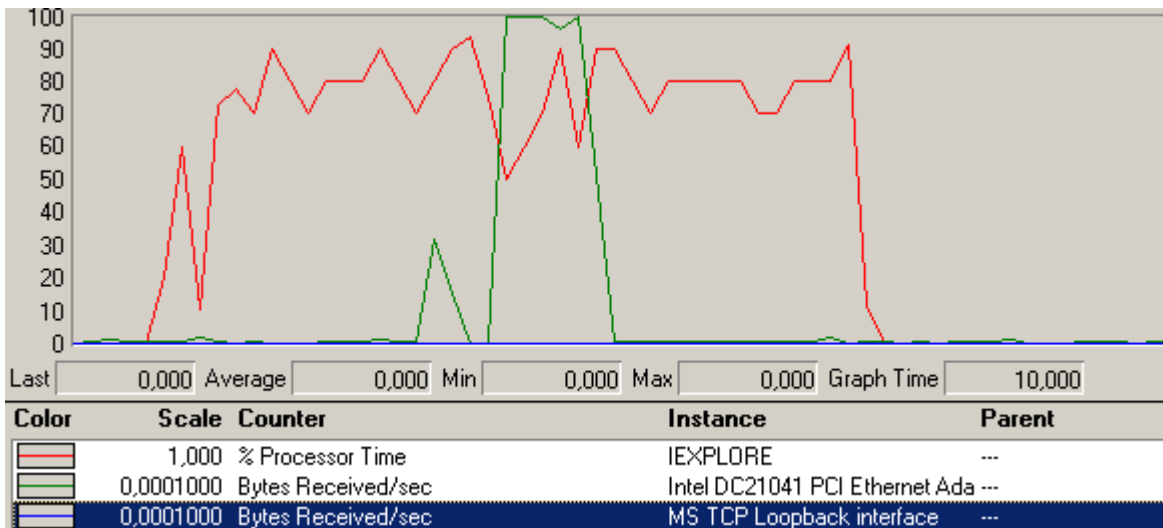


Abbildung 20: Verlauf der CPU-Aktivität (rot) und Netzwerklast (grün) beim Laden eines CR-Bildes direkt nach MSIE-Start. Erkennbar ist die Phase hoher CPU-Aktivität des IEXPLORE-Prozesses vor Beginn des Datentransfers, die bei alleiniger Betrachtung der CPU-Aktivität eine verlängerte Ladezeit des ersten Bildes vortäuscht.

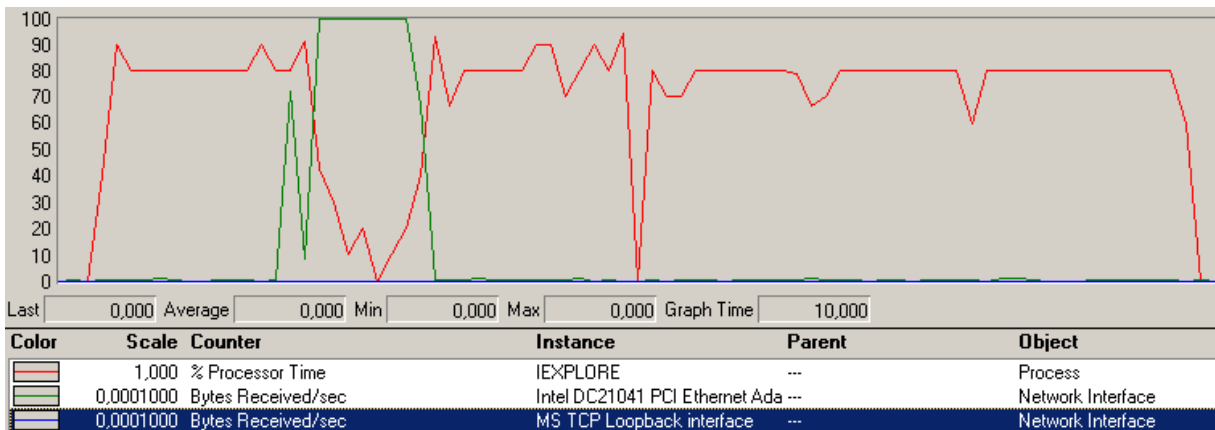
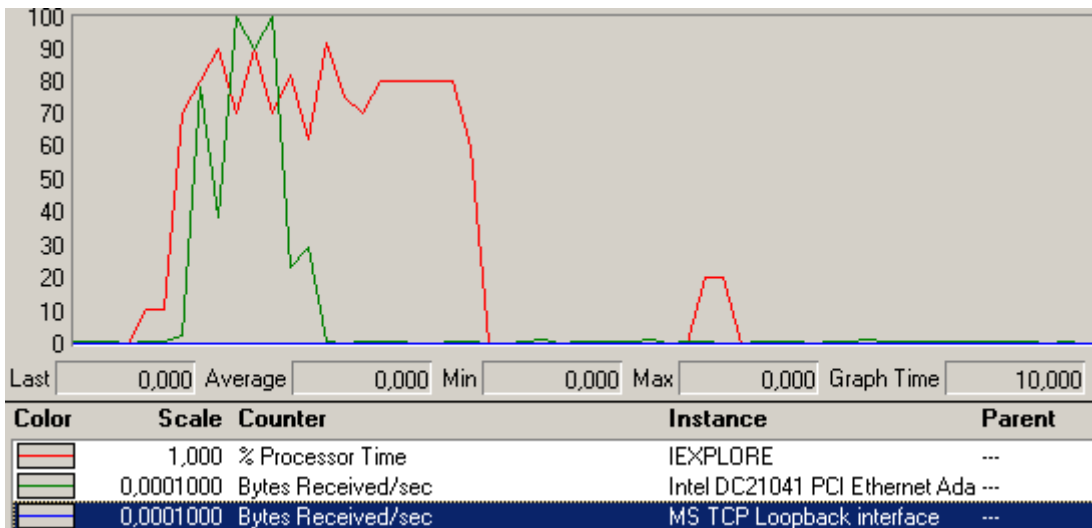
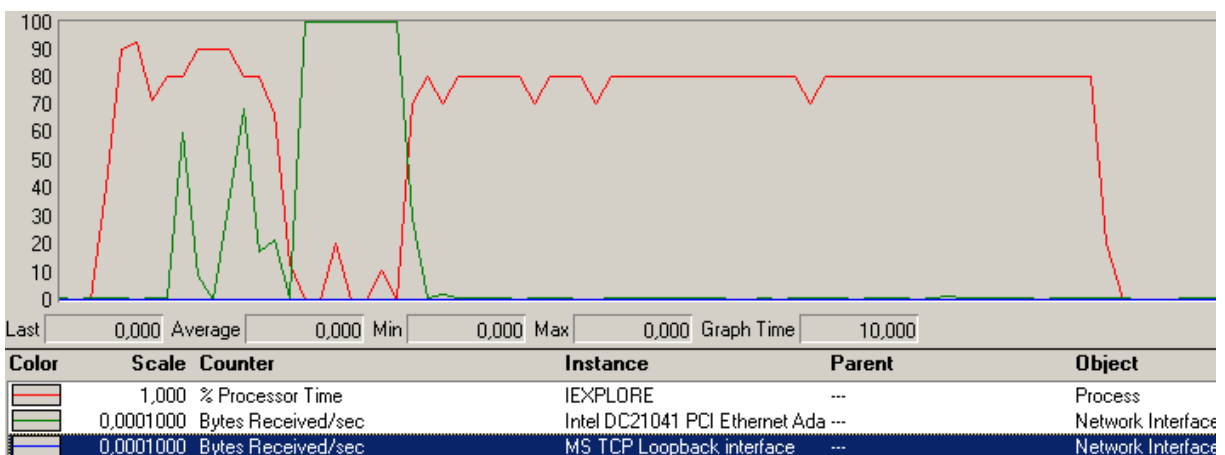


Abbildung 21: Verlauf der CPU-Aktivität (rot) und Netzwerklast (grün) beim Laden eines 4 x 4-Tableaus aus CT-Bildern direkt nach MSIE-Start. Erkennbar ist die Phase hoher CPU-Aktivität des IEXPLORE-Prozesses vor Beginn des Datentransfers, die bei alleiniger Betrachtung der CPU-Aktivität eine verlängerte Ladezeit des ersten Tableaus vortäuscht.



**Abbildung 22:** Verlauf der CPU-Aktivität (rot) und Netzwerklast (grün) beim Laden eines CR Bildes bei 10 Mbit/s. Bei 100 Mbit/s ist die Dauer der Netzwerklast noch kürzer im Vergleich zur Dauer der IEXPLORE-Aktivität.

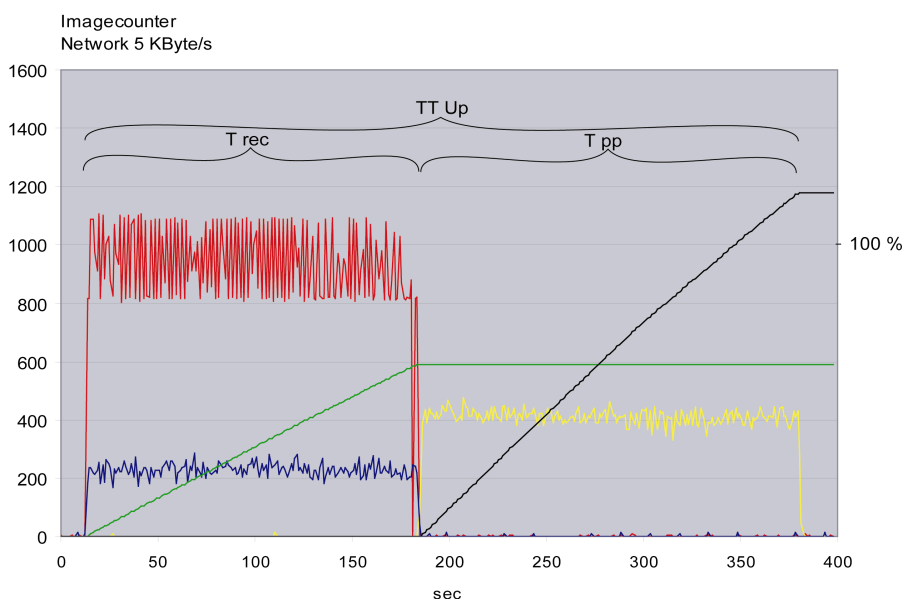


**Abbildung 23:** Verlauf der CPU-Aktivität (rot) und Netzwerklast (grün) beim Laden eines 4 x 4-Tableaus aus CT-Bildern. Auch hier hat die Zeit, in der ein signifikanter Netzwerktransfer stattfindet, nur einen kleinen Anteil an der gesamten „Ladezeit“, die durch das Plateau der IEXPLORE-Aktivität dargestellt wird.

In Bezug auf den Server selbst zeigt Abbildung 24 den Verlauf eines Hochladevorgangs für eine einzelne Studie mit 590 Bildern. Sofort nach dem Start des Hochladens auf dem Sender-PC, stieg die Netzwerkaktivität und die Prozessoraktivität des Storage Service Class Provider (S-SCP) stark an und blieb während des gesamten Vorganges auf einem hohen Niveau mit nur geringen Schwankungen. Gleichzeitig stieg die Anzahl der Dateien im „Image Cache“ des Exhibit-Servers linear an, bis alle 590 Bilder übertragen waren und blieb dann bei dieser Zahl stehen. Nach Abschluß des eigentlichen Hochladens



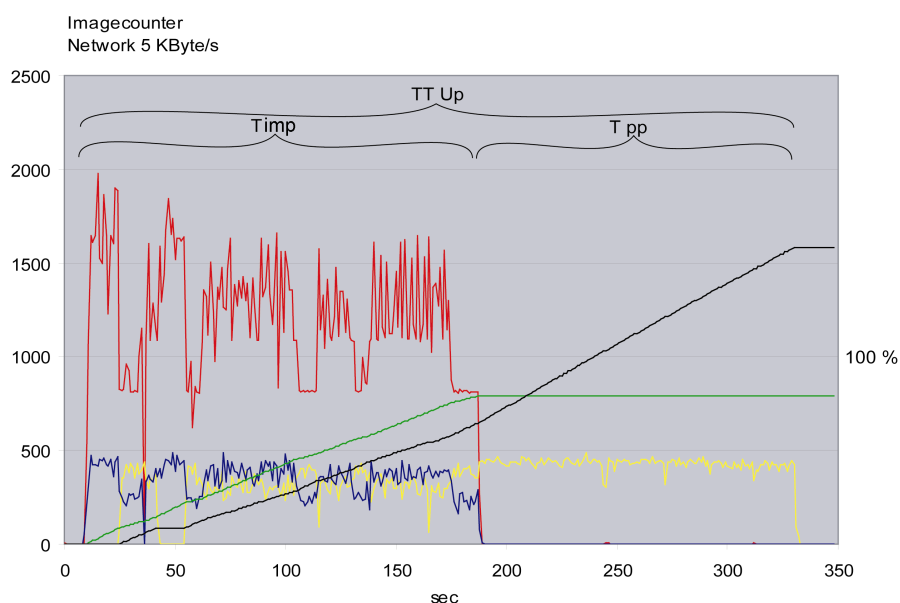
endete die Netzwerkaktivität und die Prozessornutzung des S-SCP. Dafür begann der Curator-Prozeß, die CPU zu nutzen und die Zahl der Dateien im „Web Cache“ Verzeichnis stieg linear bis auf 1.180 Dateien (590 Wavelets plus 590 JPEG-Miniaturansichten) an. Kurz nachdem die 1.180 Dateien angelegt waren, endete die Prozessornutzung des Curator-Prozesses. Daraus ergaben sich zwei verschiedene Phasen des Hochladens. Zum einen die reine Übertragung über das Netzwerk  $t_{rec}$ , zum anderen die Zeit der Nachbearbeitung  $t_{pp}$ , die zusammen die Hochladezeit TTUp ergeben.



**Abbildung 24: Beobachtung von fünf Parametern auf dem BVS-Server während eines Hochladevorgangs. Eine einzelne Studie mit 590 Bildern wurde hochgeladen. Die linke Y-Achse bezeichnet die Anzahl der Dateien in den Verzeichnissen, die rechte die Prozessornutzung. Rot: Netzwerkaktivität; Blau: Prozessornutzung des S-SCP; Gelb: Prozessornutzung des Curator; Grün: Anzahl der Dateien im „Image Cache“; Schwarz: Anzahl der Dateien im „Web Cache“ (je ein Wavelet und ein JPEG).**

Im Falle der sechs verschiedenen Studien mit zusammen 791 Bildern blieb das Verhalten im Prinzip das gleiche (Abbildung 25). Nach dem Start der Übertragung stiegen Prozessornutzung durch den S-SCP und Netzwerkauslastung auf ein höheres Niveau an als bei der Übertragung einer einzelnen Studie. Nach einiger Zeit fielen sie leicht ab, während der Curator begann, die CPU zu nutzen. Nach kurzer Zeit endete die CPU-Nutzung des Curators und die beiden anderen Leistungsindikatoren stiegen wieder auf das ursprüngliche Niveau. Im Anschluß blieben Netzwerkaktivität, CPU-Nutzung des S-SCP und des Curator hoch, schwankten aber deutlich. Gleichzeitig stieg die Zahl der Dateien in den

Verzeichnissen „Image Cache“ und „Web Cache“ stetig an. Nachdem die Netzwerkaktivität und die Aktivität des S-SCP wieder zurückgingen, nahm die Aktivität des Curator leicht zu und die Steigung der Kurve, die die Zahl der Dateien im „Web Cache“ repräsentiert, nahm zu. Beides blieb im Anschluß bis kurz nachdem die Zahl der Dateien im „Web Cache“ 1.582 erreicht hatte konstant. Daraus ergeben sich wieder zwei Phasen, zum einen die Übertragung der Bilder mit gleichzeitiger Curator-Aktivität  $t_{imp}$ , zum anderen die Phase der ausschließlichen Aktivität des Curator  $t_{pp}$ , die zusammen wieder die Hochladezeit  $TT_{Up}$  ergeben.



**Abbildung 25: Beobachtung von fünf Parametern auf dem BVS-Server während eines Hochladevorgangs. Sechs verschiedene CT-Studien mit verschieden vielen Einzelbildern (insgesamt 791 Bilder) wurden nacheinander hochgeladen. Die linke Y-Achse bezeichnet die Anzahl der Dateien in den Verzeichnissen, die rechte die Prozessornutzung. Rot: Netzwerkaktivität; Blau: Prozessornutzung des S-SCP; Gelb: Prozessornutzung des Curator; Grün: Anzahl der Dateien im „Image Cache“; Schwarz: Anzahl der Dateien im „Web Cache“.**

Abbildung 26 zeigt, daß die Gesamtzeit für das Hochladen von Bildern unabhängig davon ist, ob die Bilder nacheinander oder gleichzeitig von den drei Sender-PCs aus hochgeladen wurden. Allerdings war  $t_{imp}$  wesentlich kürzer, wenn alle drei Sender-PCs gleichzeitig Bilder übertrugen, dies wurde aber von einer längeren  $t_{pp}$  kompensiert.

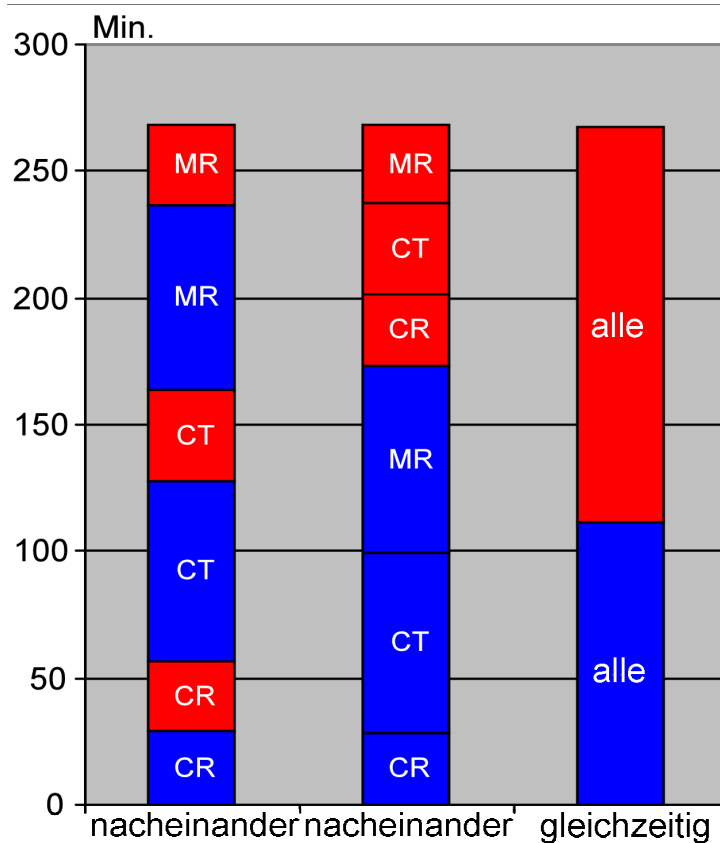
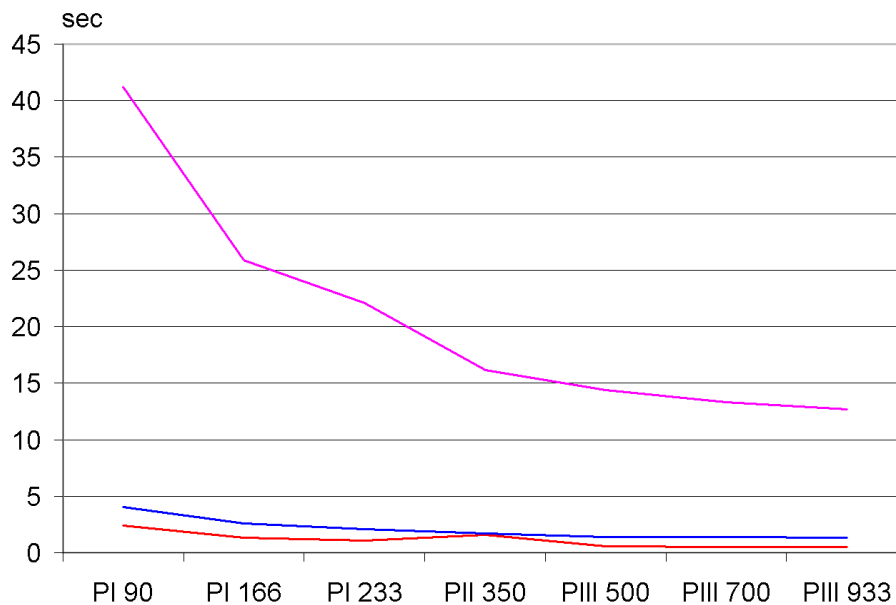


Abbildung 26: Ergebnisse des zweiten Vortests: Hochladezeit für alle Bilder der drei Sender-PCs. Die linke Spalte zeigt das Ergebnis, wenn die Bilder nacheinander gesendet werden, in der mittleren Spalte sind die gleichen Werte anders sortiert dargestellt. Rechts werden die Werte für gleichzeitiges Senden der drei Sender-PCs dargestellt. Die blauen Areale symbolisieren die Import-Zeit ( $t_{imp}$ ), die roten die Zeit für die Nachbearbeitung ( $t_{pp}$ ).

### 4.3 Klient-Hardware-Tests

Abbildung 27 zeigt, daß die für die Anmeldung am BVS benötigte Zeit mit zunehmender Prozessorleistung deutlich abnahm. Im Gegensatz dazu lag die für das Anzeigen von Miniaturansichten benötigte Zeit immer unter fünf Sekunden und wurde für das praktische Arbeiten nur unwesentlich beeinflusst. Hier angegeben sind nur die Zeiten für das erstmalige Aufrufen eines Patienten in der Miniaturansicht, das Zurückschalten zur Miniaturansicht nach dem Laden einer Serie des gleichen Patienten lag immer deutlich unter einer Sekunde.

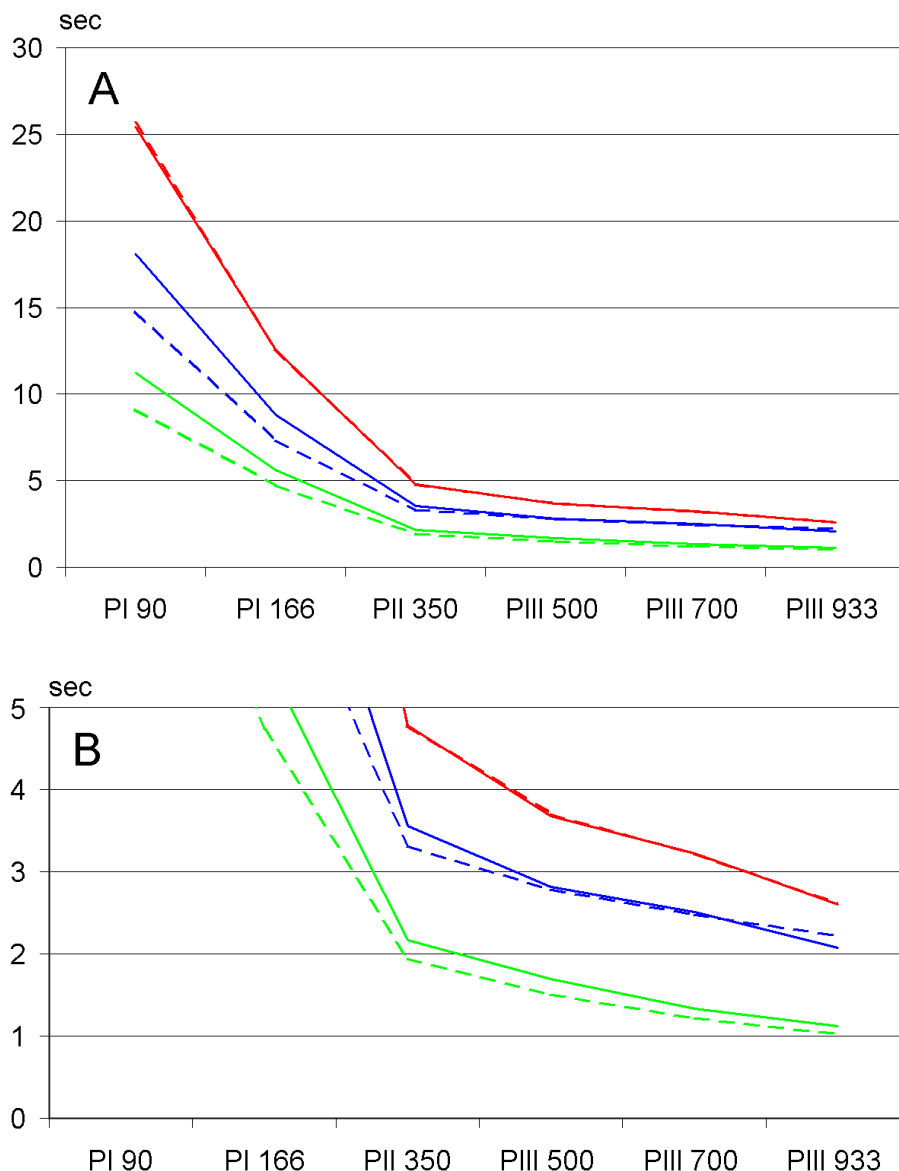


**Abbildung 27: Nicht-Bildanzeigeoperationen bei verschiedenen Prozessorgenerationen und Taktfrequenzen. Rosa: Anmeldung am System; Rot: Miniaturansicht für CT; Blau: Miniaturansicht für MR.**

Bei der Betrachtung der Bildanzeigezeiten bei allen Serien können einige allgemeine Aussagen gemacht werden (Abbildung 28 - Abbildung 32). Die Anzeigzeit für CR-Einzelbilder war unabhängig von der Klient-Konfiguration immer am kürzesten, die für 4 x 4 CT-Bilder am längsten. Von allen getesteten Einzelparametern hatte der Prozessor den größten Einfluß. Bei CT-Bildern konnte kein bedeutender Unterschied zwischen verlustfreier Übertragung und verlustbehafteter Sekundärkompression festgestellt werden.

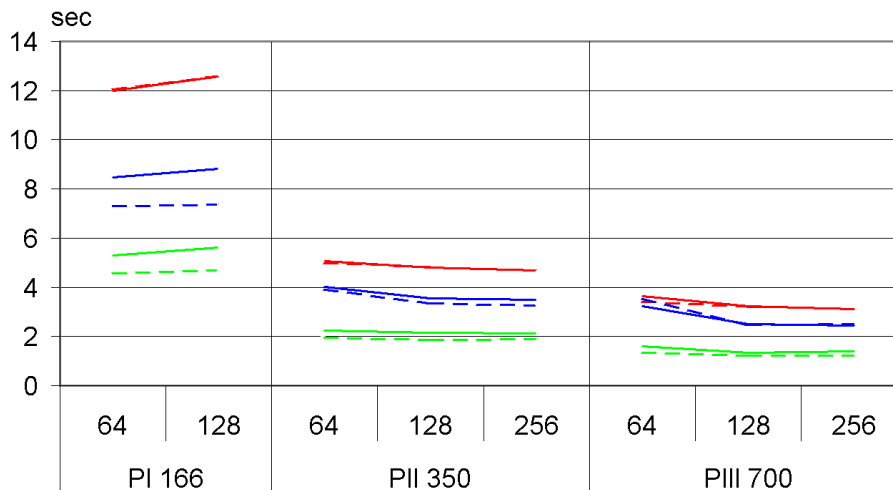
Höhere Prozessorleistung führte bei allen Bildtypen zu einer deutlichen Beschleunigung. Als Faustformel kann gesagt werden, daß eine Verdopplung der Prozessor-Taktfrequenz zu einer Halbierung der Anzeigzeit führte

(Abbildung 28). Bei allen Prozessoren mit mehr als 350 MHz lag die Anzeigzeit im Prinzip unter fünf Sekunden, bei weiterer Steigerung wurden die Zugewinne kleiner, so daß von einem asymptotischen Verhalten dieses Parameters ausgegangen werden kann. Es zeigte sich, daß verlustbehaftete Kompression bei langsameren Klientprozessoren bessere Ergebnisse lieferte als verlustfreie. Mit zunehmender Prozessorleistung nahm dieser Unterschied drastisch ab und kehrte sich im Fall von MR bei Prozessorleistungen über 500 MHz sogar um.



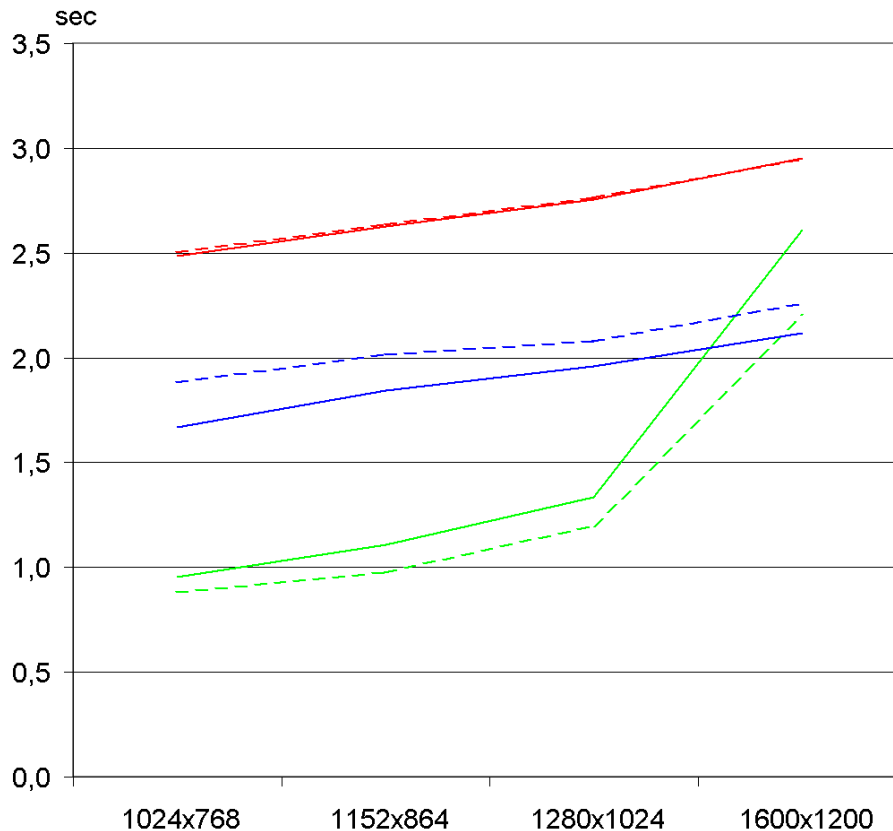
**Abbildung 28: Anzeigzeiten (in Sekunden) für verschiedene Prozessoren. B zeigt den unteren Bereich von A genauer. Grün: CR; Blau: CT; Rot: MR; durchgezogene Linie: verlustfreie Kompression; gestrichelte Linie: verlustbehaftete Kompression.**

Wie Abbildung 29 zeigt, war der Einfluß vom Arbeitsspeicher (RAM) nicht sehr ausgeprägt. Bei dem Pentium I-Prozessor führte der Einbau von mehr RAM zu einer leichten Zunahme der Anzeigezeiten, bei Pentium II- und III-Prozessoren hatte RAM einen positiven Einfluß.



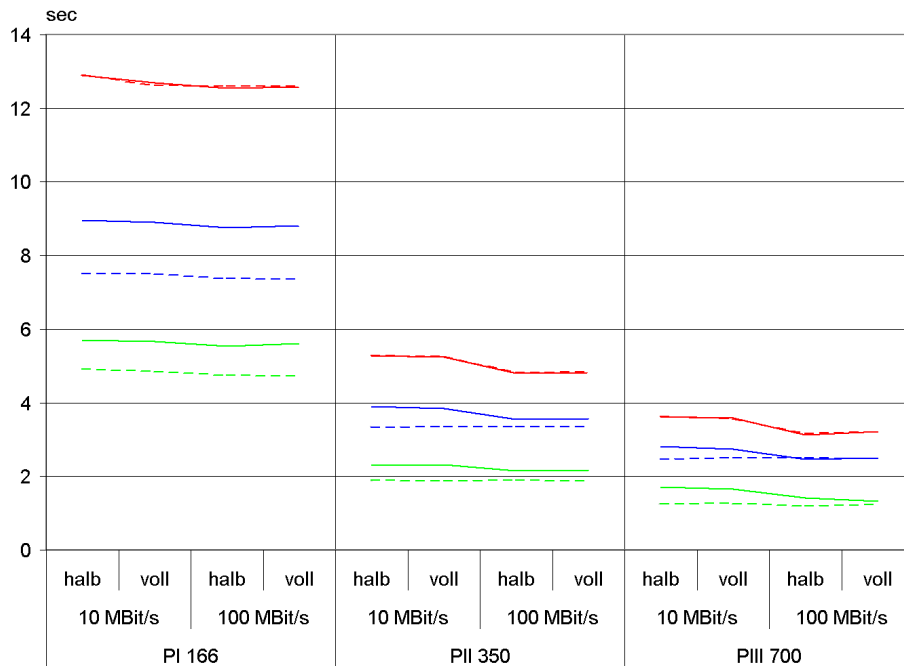
**Abbildung 29: Anzeigezeiten (in Sekunden) für verschiedene Arbeitsspeichergößen. Die drei Sektionen entsprechen den drei Referenz-PCs. Grün: CR; Blau: CT; Rot: MR; durchgezogene Linie: verlustfreie Kompression; gestrichelte Linie: verlustbehaftete Kompression.**

Höhere Bildschirmauflösungen führten zu steigenden Anzeigezeiten. Beim Pentium III mit 700 MHz und einer 8 MB Accelerated Graphics Port (AGP)-Grafikkarte bei einer Farbtiefe von 16 Bit (Abbildung 30) war dieses Verhalten bis zu einer Auflösung von 1600 x 1200 bei CT und MR linear. Im Gegensatz dazu kam es bei CR-Bildern nur bis zu einer Auflösung von 1280 x 1024 zu einem linearen Verhalten. Bei weiter steigender Auflösung auf 1600 x 1200 kam es zu einer Verdopplung der Anzeigezeit. Bei CR-Bildern war verlustbehaftete Kompression etwas schneller als verlustfreie, bei steigender Auflösung nahm dieser Vorteil zu. Wie bereits beschrieben traf für MR das Gegenteil zu, verlustbehaftete Kompression war unabhängig von Bildschirmauflösung langsamer als verlustfreie. Die Änderung der Grafikkarten und deren RAM-Ausstattung hatte nur marginalen Einfluß auf die Anzeigezeiten. Bei einer Farbtiefe von 16 Bit wurden die kürzesten Ladezeiten erreicht, bei 8 und 24 Bit Farbtiefe waren die Ergebnisse schlechter.

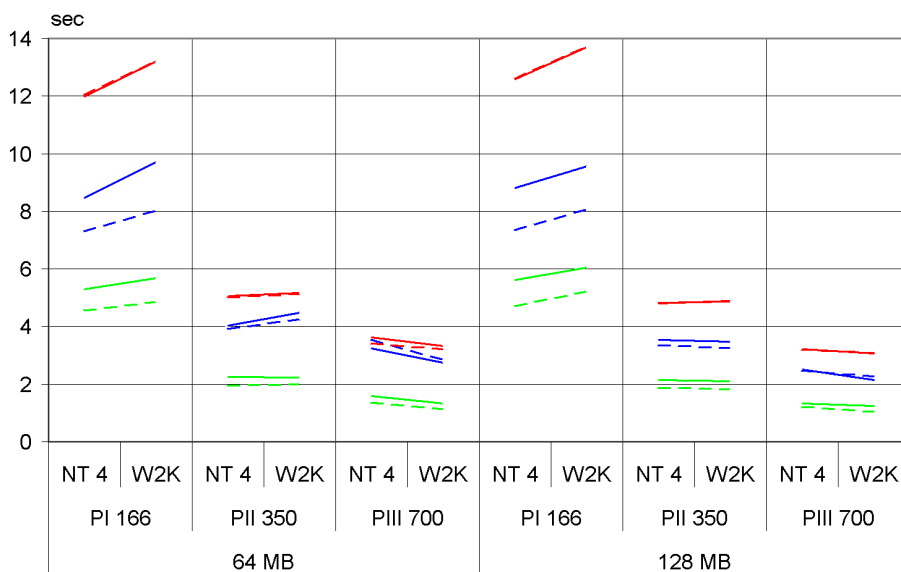


**Abbildung 30: Anzeigzeiten (in Sekunden) für verschiedene Bildschirmauflösungen (angegeben in Pixeln). Die gezeigten Ergebnisse wurden mit dem Pentium III mit 700 MHz, einer AGP-Grafikkarte mit 8 MB RAM und 16 Bit Farbtiefe gemessen. Grün: CR; Blau: CT; Rot: MR; durchgezogene Linie: verlustfreie Kompression; gestrichelte Linie: verlustbehaftete Kompression.**

Die 100 Mbit/s Netzwerkanbindung unserer Klienten brachte im Gegensatz zu einer 10 Mbit/s Anbindung eine gewisse Abnahme der Anzeigzeit bei allen Prozessorgenerationen (Abbildung 31). Weder bei 10 Mbit/s noch bei 100 Mbit/s hatte der Duplexmodus einen besonderen Einfluß. Bei verlustbehafteter Kompression war die Anzeigzeit nahezu konstant und von der Netzwerkgeschwindigkeit nicht beeinflußt. Der Unterschied zwischen den Kompressionsverfahren nahm mit steigender Netzwerkgeschwindigkeit ab und kehrte sich im Fall von MR beim 700 MHz Pentium III sogar um.



**Abbildung 31: Anzeigezeiten (in Sekunden) bei verschiedenen Netzwerkgeschwindigkeiten (Mbit/s). Die drei Sektionen entsprechen den drei Referenz-PCs. halb: Halbduplex; voll: Vollduplex; Grün: CR; Blau: CT; Rot: MR; durchgezogene Linie: verlustfreie Kompression; gestrichelte Linie: verlustbehaftete Kompression.**



**Abbildung 32: Anzeigezeiten (in Sekunden) bei verschiedenen Betriebssystemen. Die drei Sektionen entsprechen den drei Referenz-PCs mit angegebenem RAM im MB. NT4: Windows NT 4.0 Workstation; W2K: Windows 2000 Professional; Grün: CR; Blau: CT; Rot: MR; durchgezogene Linie: verlustfreie Kompression; gestrichelte Linie: verlustbehaftete Kompression.**

Abbildung 32 zeigt, daß Windows NT 4.0 auf dem Pentium I bessere Leistung erbrachte als Windows 2000, umgekehrt war Windows 2000 auf dem Pentium



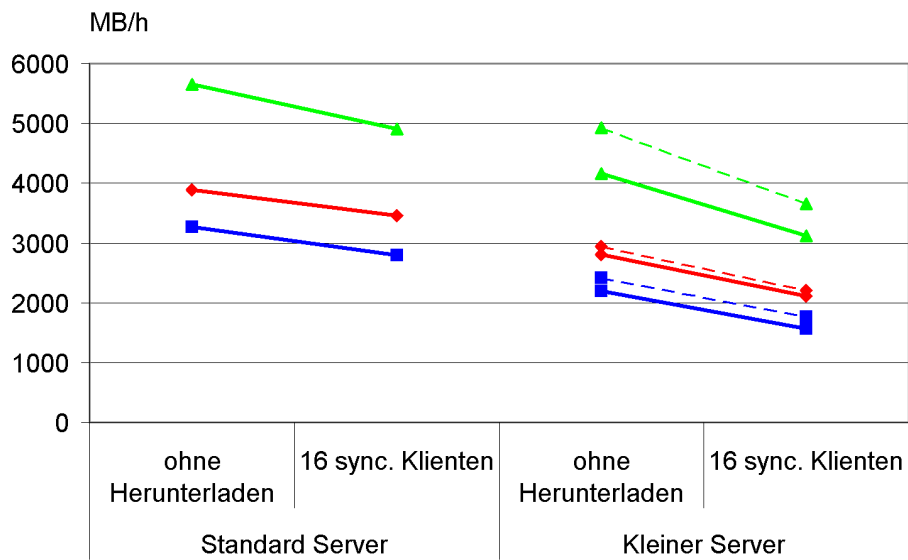
III performanter. Beim Pentium II konnte kein wesentlicher Unterschied festgestellt werden.

#### 4.4 Hochlade-Tests

Aus Abbildung 33 und Tabelle 11 geht hervor, daß von allen untersuchten Server- und Kompressionskonfigurationen die geringste Hochladekapazität für MR-Bilder erreicht werden konnte (3.270 MB/h beim Standardserver). Die Hochladekapazität des Standardserver lag mit 3.885 MB/h für CT-Bilder etwas höher, CR-Bilder konnten mit einer Rate von 5.648 MB/h angenommen werden. Der „Kleine Server“ war beim Annehmen von Bildern mit verlustfreier Primärkompression 26 % (CR), 28 % (CT) bzw. 33 % (MR) langsamer als der Standardserver. Verlustbehaftete Primärkompression verringerte diesen Abstand auf 13 % (CR), 24 % (CT) und 26 % (MR). Das gleichzeitige Herunterladen von Bildern mit 16 synchronisierten Klienten verringerte die Hochladekapazität für alle Bildtypen um rund 13 % beim Standardserver und um 26 % beim heruntergerüsteten „Kleinen Server“, bei letzterem unabhängig von der gewählten Primärkompression.

	Standard Server, verlustfreie Kompression		Kleiner Server, verlustfreie Kompression		Kleiner Server, verlustbehaftete Kompression	
	Kein Herunterladen	16 Klienten	Kein Herunterladen	16 Klienten	Kein Herunterladen	16 Klienten
CR	5648	4909	4163	3125	4923	3660
CT	3885	3454	2807	2110	2946	2210
MR	3270	2794	2197	1569	2413	1763

**Tabelle 11: Hochladekapazität in MB/h für verschiedene Bildtypen. Messungen wurden mit drei verschiedenen Serverkonfigurationen durchgeführt, jeweils ohne und mit 16 synchronisierten Klienten.**



**Abbildung 33: Hochladekapazität (in MB/h) für drei verschiedene Bildtypen. Messungen wurden für drei Server- und Kompressionskombinationen durchgeführt: Mit und ohne gleichzeitiges Herunterladen durch 16 Klienten. Grün: Hochladen von CR-Bildern; Rot: Hochladen von CT-Bildern; Blau: Hochladen von MR-Bildern; durchgezogene Linie: verlustfreie Primärkompression; gestrichelte Linie: verlustbehaftete Primärkompression.**

## 4.5 Server-Tests mit mehreren synchronisierten Klienten

### 4.5.1 Ohne Hochladen

In Anbetracht der Testergebnisse können einige generelle Aussagen getroffen werden. Es bestand immer ein Zusammenhang zwischen der Anzahl der synchron anfordernden PCs und der Anzeigzeit der Bilder. Die Anzeigzeit stieg mit zunehmender Klientenzahl praktisch linear an (Abbildung 34 - Abbildung 37). Der Hauptunterschied zwischen den Kurven ist deren Steigung und Ausgangspunkt. Bei den meisten Serverkonfigurationen blieb die Anzeigzeit selbst dann unter fünf Sekunden, wenn 16 Klienten gleichzeitig Bilder anforderten. Ausnahmen ergaben sich nur bei einer 10 Mbit/s Netzwerk-anbindung des Servers und bei verlustbehafteter Sekundärkompression bei MR-Bildern (Abbildung 37, Seite 71).

Wie Abbildung 34 zeigt, brachte die Einsparung eines Prozessors einen leichten Anstieg der Anzeigzeiten mit sich, dies gilt vor allem für CR-Bilder und ist bei CT- und MR-Serien weniger ausgeprägt. Die Kurven für einen bzw. zwei Prozessoren verlaufen praktisch parallel und entfernen sich auch bei steigender Klientenanzahl nicht voneinander.

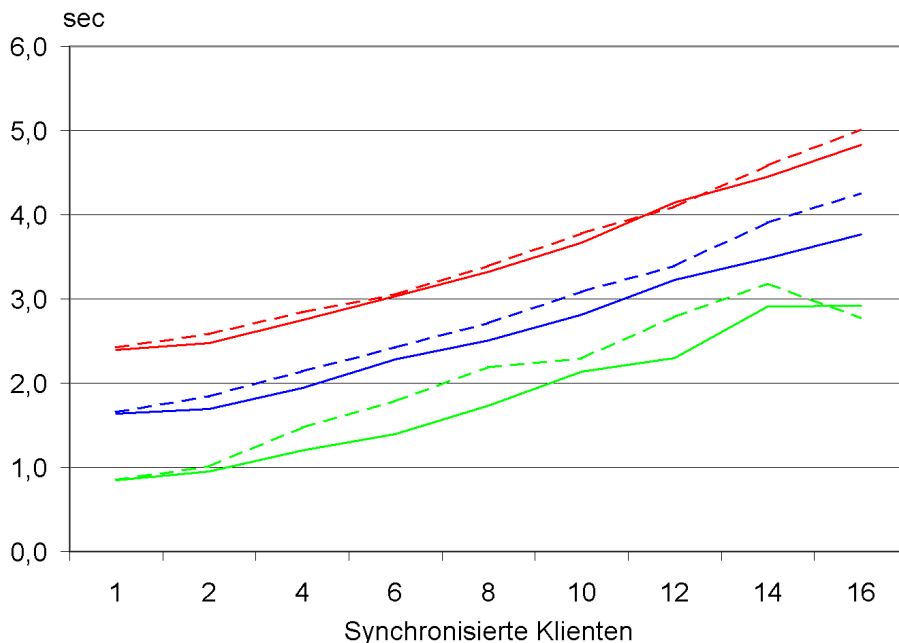
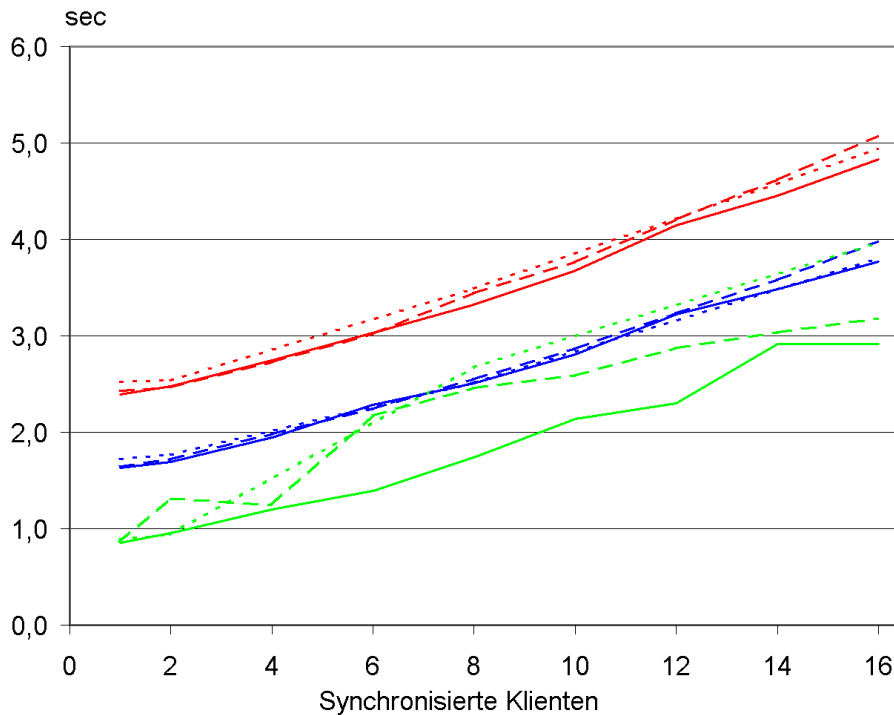


Abbildung 34: Anzeigzeiten (in Sekunden) für verschiedene Server-Prozessorzahlen und synchronisierte Klienten. Grün: CR; Rot: CT; Blau: MR; Durchgezogene Linie: zwei Prozessoren; Gestrichelte Linie: ein Prozessor.

Das Herabsetzen des Arbeitsspeichers (Abbildung 35) führte vor allem bei CR-Bildern zu einem Anstieg der Anzeigezeiten, allerdings wurde dies erst bei sechs und mehr Klienten deutlich. Für CT- und MR-Serien war dieser Unterschied vernachlässigbar.

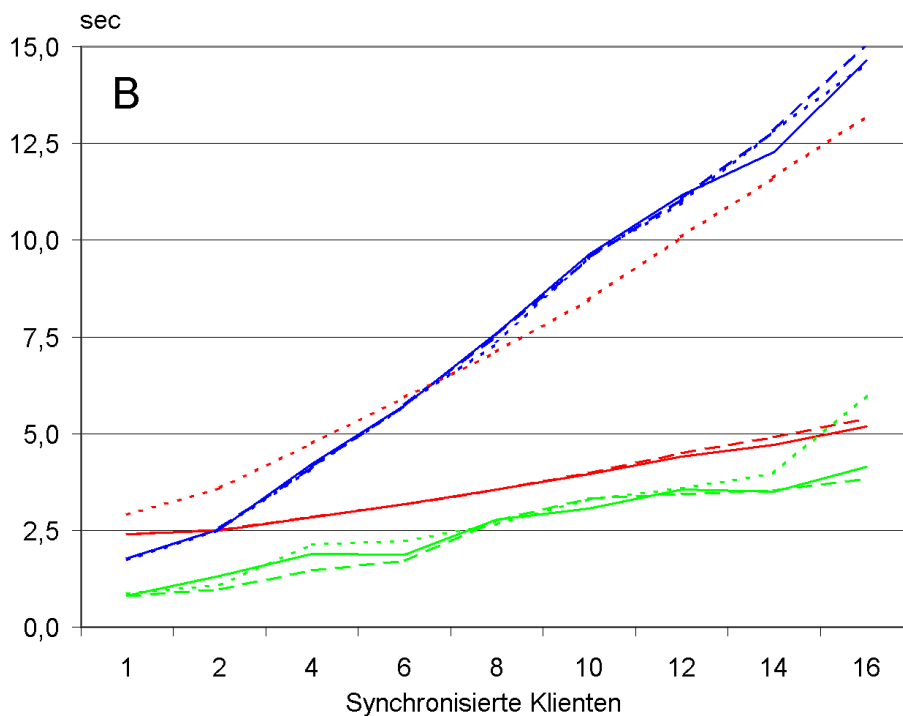
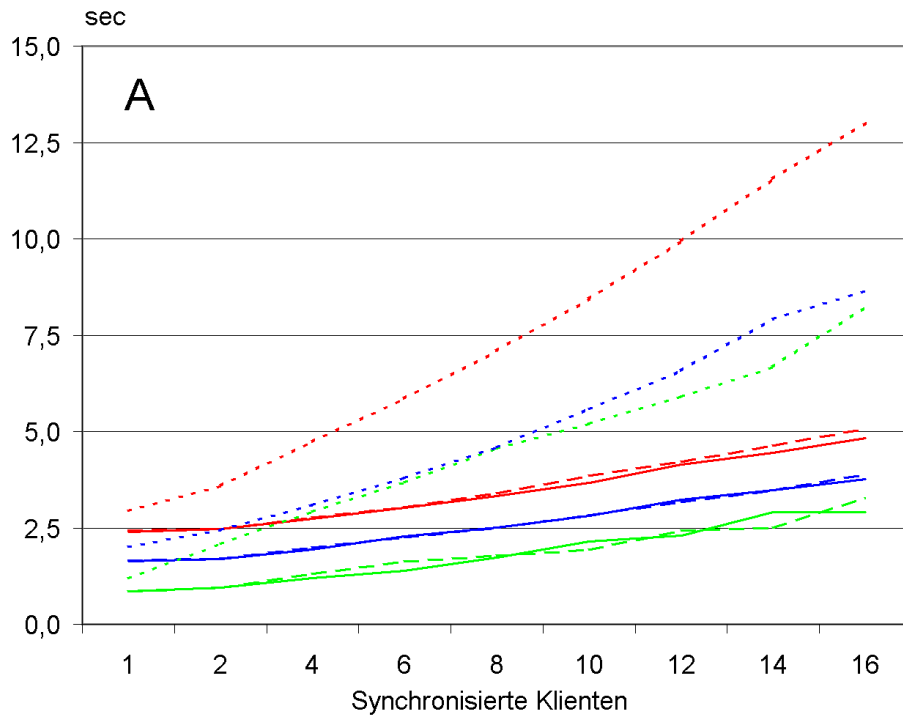


**Abbildung 35: Anzeigezeiten (in Sekunden) für verschiedene Server-Arbeitsspeicher-Konfigurationen und synchronisierte Klienten. Grün: CR; Rot: CT; Blau: MR; Durchgezogene Linie: 1 GB RAM; Gestrichelte Linie: 512 MB RAM; Gepunktete Linie: 128 MB RAM.**

Beim Vergleich der verschiedenen Netzwerkanbindungen des Servers (Abbildung 36A) kann zwischen der 200 Mbit/s- und der 100 Mbit/s-Einstellung kaum ein Unterschied ausgemacht werden. Im Gegensatz dazu verschlechterten sich die Anzeigezeiten bei der Verwendung einer 10 Mbit/s-Anbindung wesentlich. Die Steigungen und Positionen der Kurven für CR- und MR-Bilder waren ähnlich, demgegenüber zeigt die Kurve für CT längere Anzeigezeiten und eine größere Steigung. Bereits bei 6 synchronisierten Klienten stiegen die Anzeigezeiten für CT-Bilder über die 5-Sekunden-Grenze. Diese wurde bei CR und MR in diesem Szenario erst bei 10 synchronisierten Klienten erreicht.

Bei der Anwendung von verlustbehafteter Sekundärkompression (Abbildung 36B) hatten Änderungen der Netzwerkanbindung keinen signifikanten Einfluß auf die Anzeigezeiten von CR- und MR-Bildern. Der Vergleich von verlustfreier

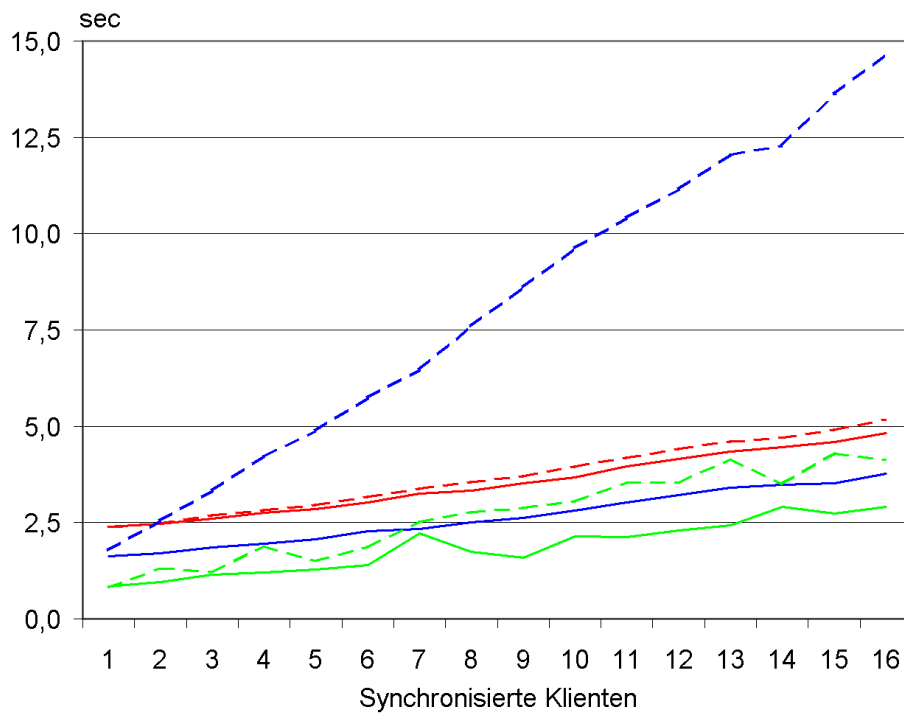
und verlustbehafteter Sekundärkompression ergibt folgendes. Bei CR-Bildern näherten sich die Kurven bei verlustbehafteter Sekundärkompression einander an und die Anzeigezeiten verlängerten sich bei 200 Mbit/s und 100 Mbit/s, wurden bei 10 Mbit/s allerdings kürzer. Also werden CR-Bilder bei 10 Mbit/s-Anbindung unter Verwendung verlustbehafteter Sekundärkompression schneller verteilt. Dies ist das einzige Szenario im Rahmen unserer Tests, in dem verlustbehaftete Sekundärkompression kürzere Anzeigezeiten lieferte. Bei MR-Tableaus bedeutet verlustbehaftete Sekundärkompression deutlich längere Anzeigezeiten und steilere Kurvenanstiege bei zunehmender Klientenzahl. Bei CT-Bildern hatte die Sekundärkompression praktisch keinen Einfluß.



**Abbildung 36A, B: Anzeigzeiten (in Sekunden) für verschiedene Server-Netzwerkverbindungen und synchronisierte Klienten. A: verlustfreie Kompression; B: verlustbehaftete Kompression. Grün: CR; Rot: CT; Blau: MR; Durchgezogene Linie: 200 Mbit/s; Gestrichelte Linie: 100 Mbit/s; Gepunktete Linie: 10 Mbit/s.**

Die Kurven in (Abbildung 37) zeigen die Unterschiede zwischen verlustfreier und verlustbehafteter Sekundärkompression. Der Einfluß von verlustbehafteter Sekundärkompression auf MR-Bilder war sehr deutlich. Die Kurve hatte eine

deutlich steilere Steigung und erreichte bei 16 synchronisierten Klienten eine Anzeigzeit von 14 Sekunden, im Gegensatz dazu blieb sie bei verlustfreier Sekundärkompression auch in diesem Fall unter fünf Sekunden. Bei CR-Bildern zeigte sich ein ähnliches Phänomen, allerdings war der Unterschied nicht so ausgeprägt. Bei CT-Bildern konnte praktisch kein Unterschied festgestellt werden.



**Abbildung 37: Anzeigzeiten (in Sekunden) für verschiedene Kompressionstypen und synchronisierte Klienten mit der Standardserverkonfiguration. Grün: CR; Rot: CT; Blau: MR; Durchgezogene Linie: verlustfreie Kompression; Gestrichelte Linie: verlustbehaftete Kompression.**

#### **4.5.2 Mit Hochladen**

Die in Abbildung 38 dargestellten Ergebnisse zeigen, daß die Anzeigezeiten für alle Bildtypen bei 16 synchronisierten Klienten ohne gleichzeitiges Hochladen von Bildern unter 5 Sekunden lagen. Während des Hochladens stiegen die Anzeigezeiten allerdings wesentlich über diese Grenze. Das Hochladen von MR-Bildern hatte den geringsten negativen Einfluß auf die Anzeigezeiten, CT-Bilder hatten einen größeren Einfluß und während des Hochladens von CR-Bildern zeigte sich die größte Verschlechterung der Anzeigezeiten.

Der Unterschied in Anzeigezeiten zwischen Standardserver und dem „Kleinen Server“ war beim Herunterladen von CR-Bildern am deutlichsten (Abbildung 38A). Dies war wesentlich weniger ausgeprägt beim Herunterladen von CT-Bildern (Abbildung 38B) und kehrte sich unter bestimmten Umständen bei MR-Bildern sogar um (Abbildung 38C).

Beim Vergleich zwischen verlustfreier und verlustbehafteter Primärkompression wird deutlich, daß die verlustbehaftete Konfiguration in den meisten Fällen geringfügig besser abschneidet und die Anzeigezeiten um bis zu zwei Sekunden verkürzt. In manchen Fällen, namentlich beim Herunterladen von CT- und MR-Bildern, während gleichzeitigem Hochladen von CT- und MR-Bildern, war der „Kleine Server“ mit verlustbehafteter Primärkompression dem Standardserver überlegen. Beim Hochladen von CR-Bildern stellte sich dieser Effekt unter keinen Umständen ein.



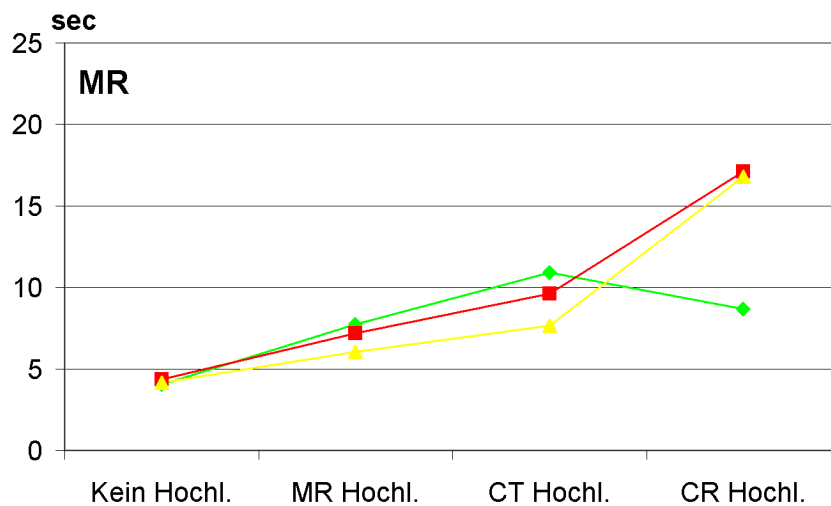
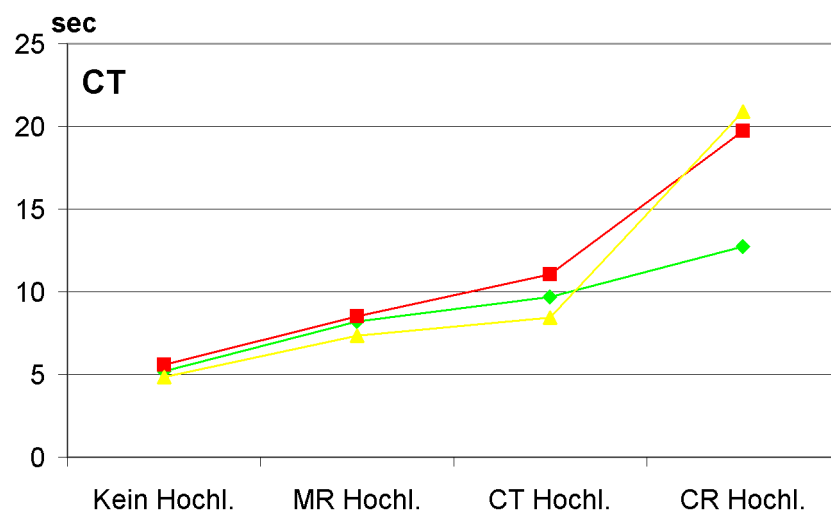
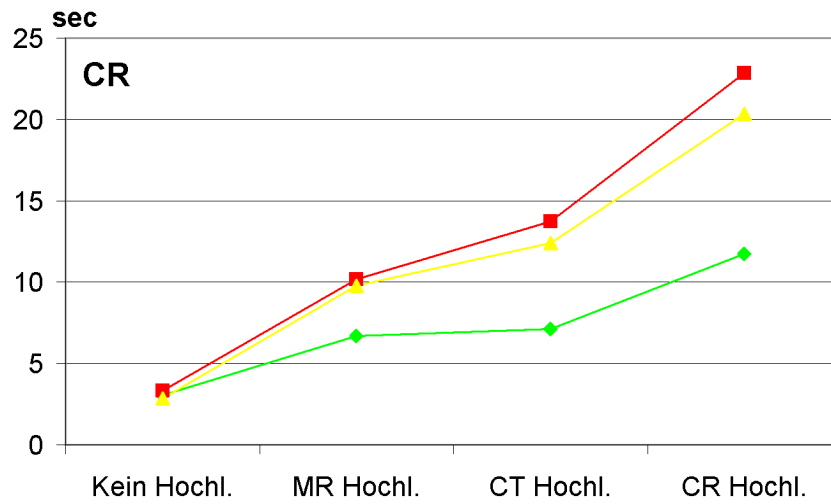
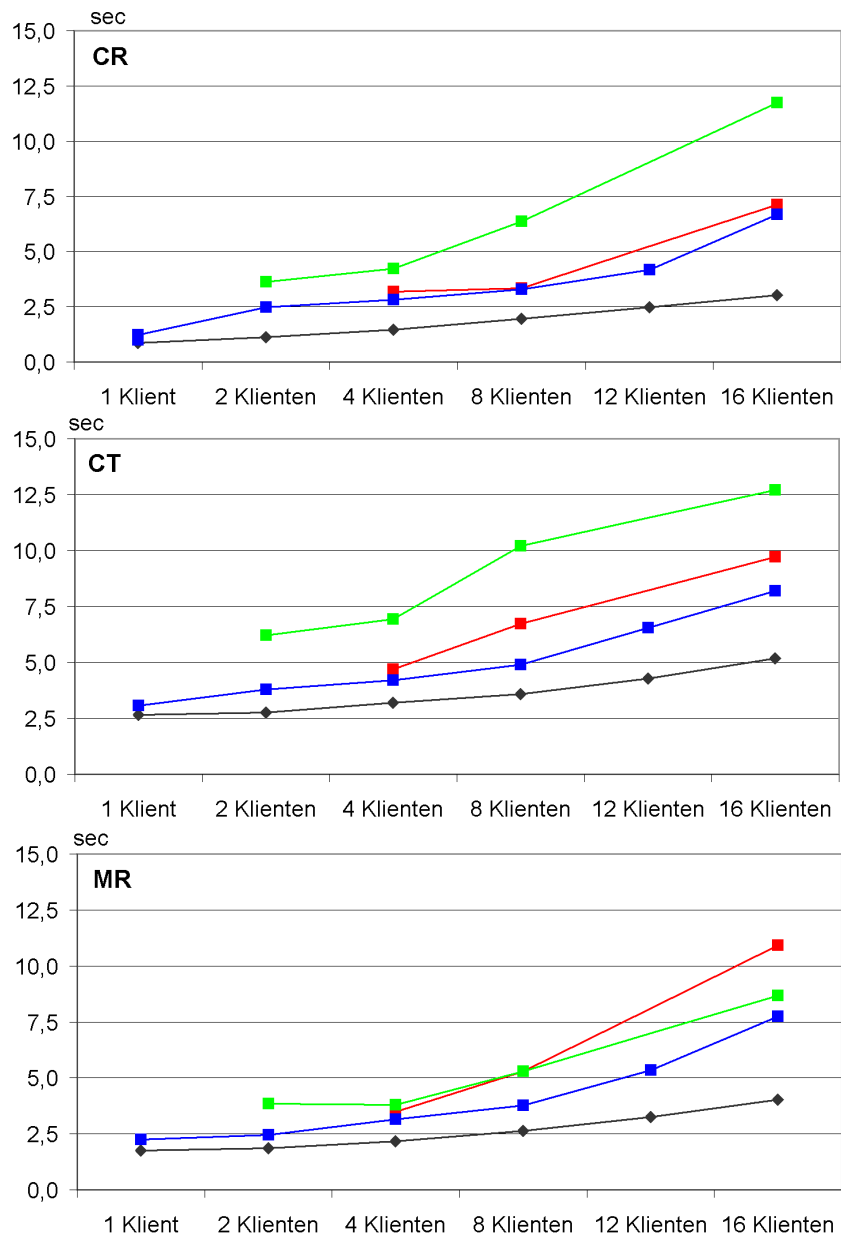


Abbildung 38: Anzeigzeiten (in Sekunden) für drei verschiedene Bildtypen mit 16 synchronisierten Klienten. Messungen wurden ohne Hochladen und während des Hochladens verschiedener Bildtypen durchgeführt. Grün: Standardserver - verlustfreie Primärkompression; Rot: Kleiner Server - verlustfreie Primärkompression; Gelb: Kleiner Server - verlustbehaftete Primärkompression.

Die Ergebnisse in Abbildung 39 bestätigen diese Feststellungen. Auch bei verschieden vielen synchronisierten Klienten hatte das Hochladen von MR-Bildern den geringsten Einfluß auf die Anzeigezeiten und das Hochladen von CR-Bildern den größten. Anzeigezeiten für CR-Bilder unterhalb von fünf Sekunden konnten unabhängig vom hochgeladenen Bildtyp eingehalten werden, solange vier oder weniger Klienten synchron Bilder abriefen (Abbildung 39A), bei MR-Bildern wurde die Fünf-Sekunden-Grenze erst bei acht Klienten überschritten (Abbildung 39C). Bei CT-Bildern lag die Anzeigezeit bis vier Klienten nur dann unter fünf Sekunden, wenn CT- oder MR-Bilder hochgeladen wurden, bei gleichzeitigem CR-Hochladen wurden keine Anzeigezeiten unter fünf Sekunden erreicht.



**Abbildung 39: Anzeigzeiten (in Sekunden) für drei verschiedene Bildtypen mit verschieden vielen synchronisierten Klienten. Messungen wurden ohne Hochladen und während des Hochladens verschiedener Bildtypen durchgeführt. Nur der Standardserver mit verlustfreier Primärkompression wurde verwendet. Schwarz: Kein Hochladen; Grün: Hochladen von CRs; Rot: Hochladen von CTs; Blau: Hochladen von MRs.**

## 5 Diskussion

Frühe Publikationen im Bereich webbasierte Bildverteilungssysteme berichten vorwiegend über Lösungen, die an verschiedenen akademischen Institutionen intern entwickelt wurden [2, 25, 26, 61, 45]. Später wurden auch Lösungen bewertet, die von der Industrie entwickelt wurden [5, 7]. In den meisten dieser Publikationen wird die Architektur der Systeme eingehend beschrieben und die grundsätzliche Nutzbarkeit der Systeme diskutiert. Aus diesem Grund sind die Vor- und Nachteile der webbasierten Bildverteilung gut dokumentiert [1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15, 17, 18, 19, 20, 23, 28, 29, 30, 32, 36, 38, 40, 41, 43, 44, 45, 50, 51, 53, 56, 60, 63, 65, 71, 76]. Einige wenige Autoren haben die Anforderungen an die Leistungsfähigkeit der Klienten analysiert [5, 7]. Diese Untersuchungen basierten jedoch zu großen Teilen auf wenigen Meßdurchläufen und auf keinem standardisierten Verfahren. Messungen wurden zumeist von Hand mit Stoppuhren durchgeführt. Bisher existieren darüber hinaus keinerlei Ergebnisse bezüglich des Verhaltens eines Bildverteilungsservers unter wirklichkeitsnahen Bedingungen mit gleichzeitigen Zugriffen von mehreren Arbeitsplätzen und während des Hochladens von Bildern. Dies liegt daran, daß diese Bedingungen äußerst schwer zu simulieren und zu standardisieren sind und daß Messungen durch den großen Zeitaufwand und die große Zahl der nötigen Wiederholungen praktisch nicht mit menschlichen Beobachtern durchgeführt werden können. Die in Methode 3 beschriebene Software erlaubt es uns erstmalig, diese Probleme zu umgehen und Messungen mit einer großen Zahl von Klienten und vielen Wiederholungen durchzuführen.

Es ist schwer zu definieren, welche Anzeigezeiten akzeptabel sind [2, 5, 28]. Unsere Erfahrung und die Informationen, die wir von praktizierenden Ärzten erhielten, legen nahe, daß eine Anzeigezeit von 5 Sekunden für CR-Einzelbilder und für CT- und MR-Tableaus mit 16 Bildern akzeptiert werden kann. Um unsere Ergebnisse einzuordnen, gingen wir von dieser auch in der Literatur immer wieder angegebenen maximalen Wartezeit von 5 Sekunden aus [5, 25, 57].

## 5.1 Grundsätzliches zum Studienaufbau

Im Rahmen eines webbasierten Bildverteilungssystems ist zu beachten, daß Kommunikation mit und relevanter Ressourcenverbrauch auf dem Server nur dann stattfindet, wenn Daten über das Netz übertragen werden. Dies ist zurückzuführen auf den Grundsatz des Internets, daß Datenübertragungen nur dann durchgeführt werden, wenn diese tatsächlich nötig sind. Im WWW ist ein einzelner Server regelmäßig dafür zuständig, seine Inhalte für die ganze Welt bereitzustellen. Dies ist nur möglich, wenn Verbindungen so kurz wie möglich gehalten werden, um Ressourcen auf dem Server zu schonen. Aus diesem Grund treten praktisch nur tatsächlich gleichzeitig stattfindende Anforderungen und Datentransfers zueinander in Konkurrenz. Wird ein Bild auf einem Klienten angezeigt, findet, außer beim Nachladen von Daten im Rahmen von Zoom-Vorgängen, kein Datentransfer und keine signifikante Ressourcennutzung auf dem Server mehr statt. Die Ergebnisse der Vortests haben zudem gezeigt, daß der Datentransfer nur einen kleinen Teil der Anzeigezeit ausmacht, die wiederum unter 5 Sekunden liegt (Abbildung 22 und Abbildung 23, Seite 56).

Selbst in einer großen BVS-Installation werden deshalb selten mehrere Klienten wirklich gleichzeitig Daten anfordern. Allerdings wird dies während der Kernarbeitszeit manchmal vorkommen. Dennoch ist davon auszugehen, daß auch in großen Institutionen nicht mehr als 16 Rechner gleichzeitig Daten anfordern.

In der Literatur wurde mehrfach untersucht, wie viele Zugriffe pro Stunde zu Höchstlastzeiten auftreten. McEnery et al. berichten, daß am Anderson Cancer Research Center in 69 Tagen 1.300 Studien in der Hauptzeit zwischen 10:00 und 11:00 Uhr abgerufen wurden, was durchschnittlich 18,8 Studien pro Stunde entspricht [44]. Teichgräber et al. geben an, daß der Nutzungshöhepunkt am Campus Virchow der Charité um 17:00 Uhr liegt und daß innerhalb von 120 Tagen 6.500 Studien in dieser Stunde abgerufen wurden, also durchschnittlich 54,2 Studien pro Stunde [64]. Beide legen einen stetigen Zuwachs in der Nutzung des Bildverteilungssystems dar. Aufgrund dieser Zahlen kann davon ausgegangen werden, daß auch zu Höchstlastzeiten selten mehr als zwei Rechner gleichzeitig Bilder anfordern.

## **5.2 Entwicklung der Meßmethodik**

Die einfachste Methode, um Ergebnisse zu erzielen, ist die Auswertung der Leistungsindikatoren, wie dies in Methode 1 beschrieben wurde. Komplexere, sichere Methoden entnehmen der Bildschirmausgabe der Anzeigeapplikation Informationen, wie ein menschlicher Benutzer dies tun würde. In diesem Szenario simuliert ein automatisierter Prozeß Eingaben über Tastatur und Maus und wertet die Bildschirmausgabe aus. Eingaben können dazu durch die Überbrückung einer Maustaste simuliert werden, während die Auswertung der Bildschirmausgabe mit Hilfe von Phototransistoren geschehen kann, wie dies in Methode 2 durchgeführt wurde. Darüber hinaus bietet MS Windows 2000 die Möglichkeit, Maus- und Tastaturoperationen innerhalb des Systems zu simulieren und die Farbe bestimmter Bildschirmpunkte auszuwerten. Eine Applikation kann so gegenüber einer anderen auf dem gleichen Rechner laufenden Applikation einen menschlichen Benutzer simulieren. Dies machte sich Methode 3 zunutze, um das BVS-Klient-Applet zu steuern und das Eintreffen von Bildern zu detektieren. Auch das in Methode 1 und 2 verwendete Tool Automouse basiert auf einem ähnlichen Verfahren, erlaubt aber keine Zeitmessungen.

### **5.2.1 Qualität der Ergebnisse**

Alle geprüften Methoden können verwendet werden, um gute Ergebnisse zu erzielen. Die Standardabweichung liegt bei allen evaluierten Methoden im akzeptablen Bereich, wobei die Standardabweichung nicht nur von der Methode, sondern vor allem auch vom betrachteten BVS abhängig ist. Allerdings erwies sich Methode 3 auch in dieser Hinsicht als überlegen. Zwischen den einzelnen Methoden bestehen gewisse Differenzen in den absoluten Ergebnissen. Diese sind jedoch im Vergleich zu den Anzeigzeiten gering und bleiben immer unter 10 %. Dabei ist zu beachten, daß das BVS selbst auch zur Standardabweichung beiträgt. Sollen verschiedene Systeme verglichen werden, so empfiehlt es sich, eine Methode auszuwählen und für alle Systeme beizubehalten. Ist dies nicht möglich, so können auch Vergleiche zwischen verschiedenen Methoden angestellt werden, solange bei der Evaluation der Ergebnisse die Unterschiede zwischen den Methoden bedacht werden.

Die Tatsache, daß die Standardabweichung bei CR- und MR-Tableaus deutlich geringer ist als bei CR-Einzelbildern erklärt sich dadurch, daß sich die Tableaus aus vielen Bildern zusammensetzen, die einzeln angefordert werden. Innerhalb eines Tableaus kommt es bereits zu einer Kompensation von langsameren und schnelleren Anzeigzeiten für einzelne Bilder.

Methode 1b ist nicht empfehlenswert, da sie eine hohe Netzwerkbelastung verursacht und damit Netzzugriffe verlangsamt, wie es sich in den längeren Anzeigzeiten für CT und MR zeigt. Die Tatsache, daß Methode 1b bei CR-Einzelbildern schnellere Ergebnisse liefert, jedoch bei CT und MR langsamere, ist konsistent mit den anderen Ergebnissen. Es wurde immer wieder gefunden, daß CT- und MR-Tableaus wesentlich empfindlicher auf Netzwerkbelastung reagieren. Dies ist dadurch erklärbar, daß beim Laden der Tableaus für jedes Bild eine Verbindung aufgebaut werden muß, während bei CR-Einzelbildern eine einzelne Verbindung ausreicht. Gerade der Aufbau einer Verbindung bindet aber auf Klient und Server Ressourcen und kostet Zeit, während derer keine Nutzdaten übertragen werden.

Methode 1 liefert wahrscheinlich deshalb langsamere Anzeigzeiten als die anderen Methoden, weil das BVS-Klient-Applet vor oder nach dem erfolgreichen Herunterladen und Anzeigen des Bildes noch weitere Maßnahmen ergreift, wie zum Beispiel das Herunterladen von Patienten- und Studieninformationen. Methode 1 kann nicht feststellen, was das BVS-Klient-Applet tut und führt jede von diesem gezeigte Aktivität auf den Lade- und Anzeigevorgang zurück, auch wenn das Bild bereits vollständig angezeigt wird. Abgesehen davon hat das ständige Auslesen der Leistungsindikatoren und Protokollieren auf Festplatte einen Einfluß auf die Performanz. Für die Messungen muß ein besonderer Prozeß laufend die Aktivität der CPU beobachten und berechnen, welcher Prozeß welchen CPU-Anteil verwendet. Im Gegensatz dazu sind die in Methode 3 ausgewerteten Bildschirmpixel einfache Abfragen von Registern. Methoden 2 und 3 benötigten keinerlei Festplattenaktivität auf dem Klienten, bei Methode 3 wurden die zu überwachenden Bildschirmpixel im RAM zwischengespeichert.

Methode 3 liefert geringfügig kürzere Anzeigzeiten als Methode 2, was durch die geringe Wiederholrate des Bildschirms von 72 Hz und Berechnungen

innerhalb der Grafikkarte ausgelöst sein kann. Die Phototransistoren können das Eintreffen des Bildes erst dann bemerken, wenn der Elektronenstrahl es tatsächlich auf den Bildschirm zeichnet, demgegenüber kann Methode 3 durch API-Aufrufe die Farbe eines Pixels bereits dann erkennen, wenn die Instruktion gerade erst an die Grafikkarte gesendet, aber noch nicht auf dem Bildschirm umgesetzt wurde. Nachleuchten des Bildschirms spielte keine ausgeprägte Rolle, da die Phototransistoren auch bei beleuchteten Arealen zwischen den Bildschirmwiederholungen immer wieder negative Werte lieferten.

### **5.2.2 Gebrauchsfähigkeit und Stabilität**

Gebrauchsfähigkeit und Stabilität spielen beim Vergleich der Methoden eine große Rolle. Hier beschriebene Faktoren umfassen die für die Entwicklung und Implementierung der Methode benötigte Zeit und die Zahl der Fehler, die das System ergab. Darüber hinaus wird betrachtet, wie leicht sich die Ergebnisse auswerten ließen.

Unsere Erfahrung zeigt, daß Methode 1 sehr leicht einzurichten ist, jedoch auch einige Fehler produziert. Um Tests mit Methode 1 beginnen zu können, muß ein einfaches Script in Automouse oder einem ähnlichen Tool entwickelt werden. Dies kann innerhalb weniger Minuten geschehen. Der mit Windows gelieferte Systemmonitor kann Leistungsindikatoren in CSV-Dateien protokollieren, allerdings ist die minimale Sampling-Rate der mit Windows 2000 gelieferten Version 1 Sekunde. Der mit Windows NT 4.0 gelieferte Systemmonitor ermöglicht ein kürzeres Intervall, erlaubt aber nur die Auswahl von Gruppen von Leistungsindikatoren, wodurch sich riesige Dateien und eine hohe Festplattenaktivität ergibt, die das System bremst. Im Internet sind Applikationen, die es erlauben mit großer Geschwindigkeit einzelne Leistungsindikatoren zu protokollieren, verfügbar, in unserem Fall wurde eine solche eigenständig entwickelt.

Die Schwierigkeit von Methode 1 liegt darin, festzustellen, welches Plateau und welcher Peak welchem Vorgang zuzuordnen sind. Da es keinen Feedback-Mechanismus zwischen Automouse, das die Bilder anfordert, den tatsächlich angezeigten Bildern und den Leistungsindikatoren gibt, kann diese Methode fälschlich davon ausgehen, daß ein Bild angezeigt wird, obwohl in Wirklichkeit der Ladevorgang abgebrochen wurde und kein Bild gezeigt wird. Es kann auch



sein, daß ein Ladevorgang so langsam stattfand, daß sich in den Leistungsindikatoren kein definierter Peak findet. Lediglich die Zahl der Peaks vor dem fraglichen Peak und die seit Start des Tests verstrichene Zeit dienen als Indikatoren. Beide können aber durch viele Faktoren beeinflusst werden. Unter anderem lösen manche Vorgänge nicht immer hohe CPU-Last aus, was die Anzahl der Peaks vor dem fraglichen verändern kann und Automouse sucht die Regionen, auf die es klicken soll tatsächlich auf dem Bildschirm, was verschiedenen lang dauern kann.

Um die Auswertung zu erleichtern, wurden lange Pausen von bis zu 40 Sekunden zwischen den einzelnen Vorgängen eingebaut und der Internet Explorer wurde zwischen den einzelnen Bildtypen beendet. Trotz der automatischen Analyse der so erhaltenen CSV-Dateien, die gewisse Probability-Checks durchführte, war es immer nötig, die Ergebnisse manuell zu prüfen und zu verifizieren. Zudem bremsen die langen Pausen die Meßsystematik deutlich.

Methode 2 wurde entwickelt, um jede Beeinflussung des Meßvorgangs auf den Klient und das BVS völlig zu eliminieren. In Methode 2 sind alle auf dem Klient stattfindenden Prozesse vollständig unabhängig von dem Meßvorgang. Auf der anderen Seite benötigt man für die Einrichtung dieser Methode gewisse Kenntnisse in Elektronik und direkter Ansteuerung der parallelen Schnittstelle eines PC. Alternativ müßte spezielle Hardware angeschafft werden. Zudem ist die Anzahl der zu beobachtenden Bildschirmareale durch steigende Kosten und Aufwand stark eingeschränkt. Die beobachteten Areale müssen deutliche Helligkeitsunterschiede und eine Größe von mindestens etwa 10 x 10 Pixeln aufweisen, da ein Phototransistor geringe Änderungen der Farbe nur schwer identifizieren kann. Das minimale Meßintervall dieser Methode entspricht der Bildschirmwiederholrate, was in den meisten Fällen ausreichen dürfte. Die Auswahl der Position der Phototransistoren auf dem Bildschirm ist leicht und auch die Zuordnung der Ergebnisse zu bestimmten Vorgängen stellt kein Problem dar, da der externe PC jeden Ladevorgang durch einen Mausklick startet. Somit kann der externe PC jedem Meßwert klar einen Vorgang zuordnen. Dennoch sind lange Pausen in den Automouse-Scripts notwendig, um einer Desynchronisation zwischen dem externen PC und Automouse bzw.

dem BVS-Klient-Applet vorzubeugen. Wie bei Methode 1 existiert kein Feedback-Mechanismus zwischen Automouse und dem Ladevorgang.

Ein klarer Vorteil der Methode 2 liegt darin, daß sie auch in einem Umfeld angewendet werden kann, in dem keinerlei Änderungen an dem zu evaluierenden Klienten vorgenommen werden können. Anstelle von Automouse kann ein Mensch die Bilder auswählen. Dies limitiert zwar die mögliche Zahl an Wiederholungen, die Messungen werden dennoch mit großer Genauigkeit durchgeführt. Dies kann in Bereichen nötig sein, in denen hochspezialisierte Anwendungen zum Einsatz kommen oder Gesetze oder Verträge jegliche Änderungen am System verbieten. Im medizinischen Bereich ist dies relativ häufig.

Die Verwendung einer Universal Serial Bus (USB) Charge-coupled Device (CCD)-Kamera, die den Bildschirm filmt, kann in Betracht gezogen werden, falls diese Methode im größeren Stil angewendet werden soll. Damit wären auch Messungen mit 4 x 4-Tableaus denkbar. Allerdings erlauben solche Kameras meist eine noch geringere Wiederholrate, zudem spielen Stroboskopeffekte zwischen Belichtungszeit und Bildschirmwiederholrate eine Rolle.

Methode 3 kombiniert die wesentlichen Vorteile beider Methoden und eliminiert deren Nachteile. Die Messungen werden vom gleichen System ausgeführt, das auch die Maus kontrolliert, so daß das Meßsystem im Gegensatz zu Methode 1 jederzeit weiß, welcher Vorgang ansteht. Dadurch können die Ergebnisse problemlos bestimmten Vorgängen zugeordnet werden. Auch lange Pausen werden unnötig, da ein Feedback zwischen Ladevorgang, Zeitmessung und Automatisierungstool besteht. Somit kann sofort nach Abschluß einer Messung die Vorbereitung für die nächste eingeleitet werden, was bei Methode 1 und 2 nicht möglich ist, da Automouse keine Kenntnis davon erlangen kann, daß die Messung abgeschlossen ist.

Allerdings setzt Methode 3 einige Programmierarbeit voraus. Die endgültige Version des Meßsystems, die in Klient- und Server-Applikation aufgeteilt ist, umfaßt ca. 3.000 Zeilen Visual Basic Code. Zudem müssen die Testscripte erstellt werden, die nicht nur die Reihenfolge der zu klickenden Pixel, sondern auch die bei Messungen zu beobachtenden Pixel umfassen. Diese Aufgabe nimmt einige Zeit in Anspruch, besonders, da sich durch die Synchronisation

mehrerer Klienten weitere Probleme für die Testscripte ergeben. So müssen Mechanismen gefunden werden die sicherstellen, daß kein Klient aus der Testreihe ausschert. Dies konnte insbesondere im Rahmen der Anmeldung am BVS-Server geschehen, da in diesem Fall 16 Klienten gleichzeitig die Anmeldung durchführen, was im Routinebetrieb außerordentlich unwahrscheinlich ist.

Die größte Schwierigkeit liegt jedoch darin, daß insbesondere cranielle CT- und MR-Serien auch innerhalb der Bilder schwarze Bereiche aufweisen, so daß der innerhalb des Bildes zu beobachtende Pixel mit Bedacht gewählt werden muß. Dies führte in unseren Tests dazu, daß für jedes einzelne 4 x 4-Tableau die 16 zu beobachtenden Pixel unabhängig definiert werden mußten. Bei fünf MR- und fünf CT-Tableaus ergeben sich allein daraus 160 zu definierende Pixel, die wiederum für verschiedene Bildschirmauflösungen angepaßt werden müssen. Nach der Erstellung der Testscripte liefen diese jedoch sehr zuverlässig.

Ein weiterer Vorteil der Methode 3 liegt darin, daß sie die Synchronisation mehrerer Klienten erlaubt, um große Last am Server zu simulieren. Dies ist bei Performanzmessungen auf dem Server notwendig. Während der Tests funktionierte Methode 3 mit bis zu 16 synchronisierten Klienten einwandfrei und blieb nur dann stehen, wenn innerhalb des BVS Fehler auftraten.

## **5.3 Vortests**

### **5.3.1 Tests unserer Umgebung**

Der Vortest im exklusiven Netz ergab klar, daß unsere Messungen problemlos im KliNet durchgeführt werden konnten. Es zeigte sich kein meßbarer Einfluß des Netzwerkes auf die Meßergebnisse. Aus technischen Gründen ist die Durchführung im bestehenden Netzwerk wesentlich einfacher als der Aufbau eines exklusiven Netzes, so daß alle Tests im KliNet durchgeführt wurden.

Vortests mit Methode 1 zeigten, daß der Server nicht gebremst wird, wenn mehrere Klienten Bilder unkoordiniert anfordern. Auf die Gründe für dieses Verhalten wurde unter 5.1 Grundsätzliches zum Studienaufbau (Seite 77) bereits eingegangen. Um ein Hochlast-Szenario zu simulieren, müssen also mehrere Klienten synchronisiert werden.

Des Weiteren wurden Vortests durchgeführt, um herauszufinden, ob es einen Unterschied macht, ob von den synchronisierten Klienten verschiedene oder gleiche Bilder angefordert werden. Es konnte kein relevanter Unterschied festgestellt werden. Die einzige Ausnahme bildete die verlustbehaftete Sekundärkompression von MR-Bildern. Da wir von verlustbehafteter Sekundärkompression in schnellen Netzen ohnehin abraten und diese gerade bei MR-Serien bei mehreren Klienten schnell inakzeptable Ergebnisse liefert, wurde nicht weiter darauf eingegangen. Da dies die Automatisierung deutlich vereinfachte, wurde bei allen Tests von allen Klienten das gleiche Bild gleichzeitig angefordert.

### **5.3.2 Untersuchung des BVS**

Bei der Analyse des BVS-Klient-Applet zeigte sich, daß nach dem Start, bereits bevor die Netzwerkübertragung des ersten Bildes einsetzt, hohe Aktivität des BVS-Klient-Applet aufgezeichnet wurde. Erst nach dieser Aktivität setzt die Netzwerkübertragung ein, während und nach der das BVS-Klient-Applet weiter viel CPU-Zeit verwendet. Unter Berücksichtigung des Ergebnisses, daß das BVS-Klient-Applet für das erste Anzeigen eines Bildes nach seinem Start länger braucht als für folgende Bilder oder Serien wird klar, daß das BVS-Klient-Applet bzw. die Virtual Machine vor dem Umschalten in den Anzeigemodus den dafür nötigen Code extrahieren, laden und den Bytecode kompilieren muß (Siehe 2.6 Java-Grundlagen, Seite 19). Dieser Schluß wurde von uns leider erst nach den Messungen mit Methode 1 und 2 gezogen, so daß alle vorher ermittelten Werte für das jeweils erste Bild nach dem Starten des BVS-Klient-Applet nur zwischen Methoden 1 und 2 vergleichbar sind. Bei den Tests mit Methode 3 wurde MSIE zwischen verschiedenen Bildtypen nicht beendet, somit war kein Nachladen und Kompilieren des Anzeigemoduls nötig.

Wenn nur eine Studie hochgeladen wird, kann eine klare Unterscheidung zwischen zwei Phasen des Hochladens getroffen werden. In der ersten Phase des Hochladens ( $t_{rec}$ ) werden die Bilder empfangen, während dieser Zeit ist der S-SCP-Prozess aktiv. Erst wenn alle Bilder eingetroffen sind, beginnt der Curator-Prozeß mit der Nachverarbeitung ( $t_{pp}$ ), die das Generieren der Miniaturansichten im JPEG-Format und das Erstellen der Wavelets, je nach Konfiguration unter Verwendung verlustbehafteter oder verlustfreier Primär-

kompression, umfaßt (Abbildung 24, Seite 57). Zusammen ergeben  $t_{rec}$  und  $t_{pp}$  die Hochladezeit TTUp.

Als mehrere Studien direkt nacheinander zum BVS übertragen wurden, überschritten sich der Empfang und die Nachverarbeitung. Sobald die erste Studie vollständig empfangen war, setzte der Curator-Prozeß ein und begann mit der Umwandlung der Bilder, während der S-SCP die nächste Studie empfing. Als alle Studien empfangen waren, blieb der Curator-Prozeß allein aktiv, erhielt aber mehr Ressourcen, so daß die Umwandlung beschleunigt wurde. Dies zeigt sich in der Steigung der „Anzahl Dateien im Web-Cache“ Kurve. Auch hier ist eine klare Abgrenzung zwischen zwei Phasen möglich. Der Zeitraum, in dem S-SCP und Curator aktiv sind, wird als  $t_{imp}$  definiert, um diesen von  $t_{rec}$  zu unterscheiden. Die Zeit, in der allein der Curator aktiv ist, wird weiterhin als  $t_{pp}$  bezeichnet.  $t_{imp}$  und  $t_{pp}$  ergeben zusammen die Hochladezeit TTUp (Abbildung 25, Seite 58). Da das direkt aufeinanderfolgende Hochladen mehrerer Studien das realistischere Szenario darstellt, wurde dies für die weiteren Studien gewählt. Während aller Messungen wurde die Prozessor-nutzung des S-SCP und des Curator überwacht, um die beiden Phasen unterscheiden zu können und um die Hochladezeit zu bestimmen.

In einer Routineinstallation werden Bilder in der Regel vom PACS, das in der Regel über eine leistungsfähige Hardware verfügt, oder von mehreren Modalitäten an das BVS übertragen. In manchen Fällen ergeben sich beide Szenarien. Dies führt zu einer großen Menge an Bildern, die innerhalb kurzer Zeit an den Server übertragen werden können. Der zweite Vortest war deshalb nötig, um eine Verfälschung unserer Ergebnisse durch eine Engstelle auf der Seite der sendenden PCs auszuschließen. Durch das gleichzeitige Senden von mehreren PCs aus konnte indirekt eine größere Hardware mit höherer Sendeleistung simuliert werden. Die Ergebnisse zeigen, daß die Hochladezeit konstant blieb und unabhängig davon ist, ob die drei Sender-PCs ihre Daten gleichzeitig oder nacheinander sendeten. Allerdings war bei drei sendenden PCs die  $t_{imp}$  kürzer, was von einer längeren  $t_{pp}$  kompensiert wurde. Das heißt, daß die Empfangskapazität unseres Servers beim Empfangen der Bilder von einem Sender-PC alleine nicht ausgelastet war. Da die Übertragung alleine nicht praxisrelevant ist, sondern die gesamte Hochladezeit, die das Umwandeln

der Bilder beinhaltet, berücksichtigt werden muß, wurde beschlossen, daß die Menge an Sender-PCs und deren Hardwareausstattung keinen limitierenden Faktor darstellte und daß ein einzelner Standard-PC als Sender-PC für die geplanten Untersuchungen ausreichen würde. Zudem zeigte dieses Ergebnis, daß das vorhandene BVS mit einer großen Eingabedatenmenge umgehen könnte und daß es sauber skalieren würde.

#### **5.4 Klient-Hardware-Tests**

Grundsätzlich wurde beobachtet, daß ein CR-Bild schneller geladen wird als 16 CT- oder MR-Bilder, dies bestätigt Bennett et al.'s Ergebnisse [5]. Dies ist nicht überraschend, wenn CR und CT verglichen werden, da die Datenmenge für 1 CR-Bild mit 8 MB im DICOM-Format der Datenmenge für 16 CT-Bilder mit je 0,5 MB im DICOM-Format entspricht. Da kontinuierliches Herunterladen immer schneller ist und seltener unterbrochen wird als 16 einzelne Ladevorgänge, erklärt sich die längere Anzeigzeit. Zudem müssen die 16 Verbindungen einzeln erstellt werden, was Ressourcen bindet und Zeit kostet, in der keine Nutzdaten übertragen werden. Allerdings lag auch die Anzeigzeit von 16 MR-Bildern, die mit je 0,125 MB nur ein Viertel der Datenmenge im DICOM-Format umfassen, höher. Dies zeigt, daß die Zahl der Bilder einen wesentlich größeren Einfluß hat als deren Größe. Dabei muß aber auch in Betracht gezogen werden, daß die Übertragung nicht im DICOM-Format, sondern im Wavelet-Format stattfindet, und daß das von uns untersuchte BVS-Klient-Applet immer nur die Menge an Daten herunterlädt, die es zum Anzeigen eines scharfen Bildes benötigt.

Im Hinblick auf Prozessorgeschwindigkeit ist zu erwähnen, daß Bennett et al. auch zu dem Ergebnis kam, daß eine Verdopplung der Prozessorleistung die Anzeigzeit halbiert [5]. Allerdings kam diese Gruppe zu dem Ergebnis, daß die Relation linear sei, was wahrscheinlich auf den kleineren Ausschnitt an Taktfrequenzen zurückzuführen ist, die untersucht wurden (200 - 400 MHz).

Unsere Ergebnisse zeigen, daß ein Arbeitsspeicher von 64 MB prinzipiell ausreicht, dies wird von Mitra auch als Mindestausstattung empfohlen. Obgleich dies nicht genauer untersucht wurde, wurde doch festgestellt, daß mindestens 128 MB verfügbar sein sollten, um flüssiges Arbeiten zu ermögli-

chen und hier nicht untersuchte Funktionen des BVS wie Filmdarstellung u.s.w. zu nutzen. Dies gilt insbesondere, wenn mehrere Studien hintereinander angezeigt und mit einander verglichen werden sollen. Das widersprüchlich scheinende Ergebnis, daß beim Pentium I Prozessor ein Zuwachs des Arbeitsspeichers auf 128 MB eine Verlangsamung zur Folge hatte, läßt sich durch das Alter des Mainboards erklären. Die Computerchips und BIOS-Versionen, die verwendet wurden als dieser Rechner zusammengesetzt wurde, waren nicht für mehr als 64 MB Arbeitsspeicher ausgelegt und arbeiteten suboptimal als mehr zur Verfügung stand.

Ein Vorteil des Wavelet-Verfahrens liegt darin, daß nicht die vollständige Datei geladen werden muß, um den Inhalt darzustellen (Siehe 2.5.2 Grundlagen des Wavelet-Verfahrens, Seite 17) [34, 59]. Die Menge an geladenen Daten entscheidet vor allem über die Schärfe des Bildes. Bereits nach dem Eintreffen des ersten Datenblocks kann das Bild unscharf dargestellt werden (Abbildung 5, Seite 24). Dies macht sich das BVS zu Nutze und überträgt nur die Menge an Daten die nötig ist, um das Bild auf dem entsprechenden Bildschirm scharf und ggf. verlustfrei darzustellen. Eine Vergrößerung der Bildschirmauflösung führt zu einem größeren Bildschirmfenster und damit zu einer größeren Datenmenge, die übertragen werden muß, um das Bild scharf darzustellen. Dies erklärt die Korrelation zwischen Zunahme der Anzeigezeit und höherer Bildschirmauflösung bei CT- und MR-Bildern. Die rein mathematische Korrelation zwischen Bildschirmauflösung und Anzahl an Pixeln ist exponentiell. Im Hinblick auf das nichtlineare Verhalten der Anzeigezeit bei CR-Bildern bei Auflösungen über 1280 x 1024 wird angenommen, daß die Kurve erst bei dieser Auflösung das exponentielle Verhalten annimmt. Es ist auch möglich, daß die Stückelung innerhalb der vom BVS-Server generierten Wavelet-Dateien so ausgelegt ist, daß bei geringeren Auflösungen nicht immer alle Daten geladen werden müssen, sondern beim Erreichen einer Schwelle die komplette Datei angefordert werden muß. Zudem haben verschiedene Aufnahmetypen nur bestimmte Auflösungen, die ab einer bestimmten Bildschirmgröße vollständig dargestellt werden können. Beides hat ein Plateau ab einer bestimmten Bildschirmauflösung zur Folge.

Das Ergebnis, daß bei einer Farbtiefe von 8 Bit die Anzeigzeit länger war als bei 16 Bit erklärt sich darin, daß der Klient die ankommende Farbenzahl zunächst rechnerisch auf 8 Bit reduzieren muß. In Gegensatz dazu muß bei einer Farbtiefe von 24 Bit eine wesentlich größere Datenmenge im Klient-PC bewältigt, also berechnet und an die Grafikkarte übertragen werden, was auch zu einem Anstieg der Anzeigzeit führt.

Die minimale Differenz in Anzeigzeiten zwischen der 10 Mbit/s und der 100 Mbit/s Netzwerkanbindung zeigt, daß das Netzwerk bei unseren Tests kein limitierender Faktor war. Voll- und Halbduplexeinstellungen haben keinen Einfluß, was nicht überrascht. Der Vorteil einer Vollduplexanbindung liegt darin, daß der Rechner gleichzeitig Daten senden und empfangen kann. Im Falle des Anzeigens von Bildern werden aber nur minimale Datenmengen gesendet, dafür aber relativ große Datenmengen empfangen. An dieser Stelle muß noch einmal darauf hingewiesen werden, daß lediglich die Anbindung des Klienten auf 10 Mbit/s reduziert wurde. Beim zugrundeliegenden Netzwerk handelte es sich um ein voll geschwitchtes Netz mit Gigabit Backbone, der Server war via FastEtherChannel angebunden. Bei einem generell langsameren Netzwerk wird dieses beim Versorgen mehrerer Klienten schnell zu einer Engstelle und die Anzeigzeiten steigen an, was der Grund für die von Bennett et al. gefundenen längeren Anzeigzeiten sein dürfte [5, 6].

Bei der Betrachtung der beiden Betriebssysteme zeigt sich, daß das neuere Windows 2000 Professional für ältere Hardware überdimensioniert ist, jedoch im Gegensatz zu Windows NT 4.0 die Vorteile und Optimierungen neuerer Hardware ausspielen kann, was zu Geschwindigkeitszuwächsen führt. Dies zeigt, daß auf neuen Rechnern moderne Betriebssysteme eingesetzt werden sollten, um optimale Leistung zu erzielen. Obwohl dies nicht getestet wurde, kann davon ausgegangen werden, daß Ähnliches für Windows XP gilt.

Entgegen allen Erwartungen kann die Verwendung verlustbehafteter Sekundärkompression nicht empfohlen werden. Nur bei langsameren Prozessoren unter 350 MHz beschleunigten sich die Anzeigzeiten unabhängig von den Netzwerkeinstellungen, bessere Ergebnisse wurden auch bei langsameren Netzwerken mit 10 Mbit/s und darunter gefunden. Allerdings verlängert die verlustbehaftete Sekundärkompression die Anzeigzeiten bei Prozessor-



geschwindigkeiten oberhalb 500 MHz und einem schnellen Netzwerk. Das gleiche berichten Bennett et al. [6] bei Schnittbildverfahren, allerdings setze der Effekt bereits bei 300 MHz ein. Diese Gruppe erklärte dies vor allem durch die notwendige Dekompression auf dem Klienten. Unter Einbeziehung der übrigen Ergebnisse und der Funktion des Wavelet-Verfahrens ist aber von einem wesentlich komplexeren Zusammenhang auszugehen. Die minimal mögliche Anzeigzeit wird von dem schwächsten Glied der Anzeigekette, die sich aus Server, Netzwerk und Klient zusammensetzt, bestimmt. Bei verlustbehafteter Sekundärkompression muß der Server vor dem Versenden der Bilddaten diese in verlustbehaftete Kompression umwandeln, was Zeit benötigt. Dies benötigt um so mehr Zeit, je mehr einzelne Bilder betroffen sind, was erklärt, warum gerade Schnittbildverfahren langsamer geladen werden. Da der Server der Arbeitsgruppe um Bennett [5] langsamer war als unserer, konnte der Effekt bereits bei einem Klienten mit 300 MHz und einem 10 Mbit/s Netzwerk beobachtet werden. Diese Überlegung wird von der Tatsache untermauert, daß die Anzeigzeiten bei synchronisiert anfordernden Klienten unter Verwendung verlustbehafteter Sekundärkompression noch deutlicher steigen. Verlustbehaftete Sekundärkompression kann also nur für PCs mit weniger als 300 MHz empfohlen werden oder in einem langsamen Netzwerk, also einem Wide Area Network (WAN). Nur in einer Installation, die grundsätzlich nur über langsame Netzwerke verfügt, sollte die Verwendung verlustbehafteter Primärkompression in Betracht gezogen werden. Allerdings sollte eine solche Institution auch in Hinblick auf andere Systeme die Erweiterung des Netzwerkes in Betracht ziehen. Installationen, die für Anwendungsszenarien wie Ausbildung oder Bildverteilung an externe Auftraggeber gedacht sind, bei denen also ein Großteil der Benutzer über WAN-Leitungen, ISDN oder DSL zugreift, kann auch verlustbehaftete Primärkompression verwenden, wenn die Auftraggeber dies billigen und rechtliche Vorgaben eingehalten werden.

## **5.5 Hochlade-Tests**

Bisherige Studien über webbasierte Bildverteilungssysteme konzentrierten sich vor allem auf deren Anzeigzeiten [5, 9, 10, 11], ohne den Einfluß des Hochladens oder die Menge der pro Zeit hochladbaren Bilder zu betrachten. Einzig Clark et al. haben Untersuchungen zu der Frage angestellt, wie schnell Bilder von Modalitäten zum PACS und an ein webbasiertes Bildverteilungs-

system übertragen werden [18]. Die Zahlen sind mit unseren jedoch kaum vergleichbar, da dort immer die für den Gesamttransfer über mehrere Systeme nötige Zeit betrachtet wird und nicht die Hochladekapazität des Servers.

### **5.5.1 Hochladekapazität**

Die Ergebnisse der Hochladetests ergeben, daß die Größe der einzelnen Bilder einen wesentlichen Einfluß auf die Menge an Daten hat, die pro Zeiteinheit hochgeladen werden können. Je kleiner das einzelne Bild, desto geringer die Datenmenge, die pro Zeiteinheit hochgeladen werden kann. Dies liegt wahrscheinlich daran, daß der verwendete Kompressionsalgorithmus mit größeren Bildern effektiver arbeiten kann als mit kleinen. Hinzu kommt, daß der Curator-Prozeß bei jedem Bild unabhängig von dessen Größe gewisse Maßnahmen ergreifen muß. Jedes Bild muß geladen, in der Datenbank eingetragen und in zwei Formaten abgespeichert werden. Diese Vorgänge produzieren einen Overhead, der unabhängig von der Größe des einzelnen Bildes ist. Hinzu kommt, daß wie beim Herunterladen der Bilder auch beim Hochladen viele kleine Transfers einen Overhead für das Erstellen der Verbindung verursachen, während dem keine Nutzdaten übertragen werden.

Mit unserem Standardserver bewegte sich die Hochladekapazität zwischen 2.7 und 5.5 GB/Stunde, abhängig vom Bildtyp. Unter Annahme eines durchschnittlichen Gemisches an Bildern [24], scheint eine Hochladekapazität um 4.0 GB/Stunde, also 96 GB pro Tag, realistisch zu sein. In der Literatur wird die durchschnittliche Bildproduktion sehr verschieden angegeben, die hängt vor allem von der Größe der Abteilung und der Anzahl der angebotenen Modalitäten zusammen. Die Werte werden in GB pro Tag (GB/t) oder Terabyte pro Jahr (TB/J) angegeben. Obgleich dies nicht vollständig korrekt ist, werden im Folgenden die Jahresangaben durch 365 geteilt, um eine Vergleichbarkeit herzustellen. In der Literatur finden sich Hinweise auf Produktionsraten von 3.4 GB/t [38], 6.2 GB/t [26], 8 GB/t [2], 9 GB/t [25], 19 GB/t [59], 20 bis 30 GB/t [61] bis hin zu 31 GB/t [18] und 34 GB/t [45] in einer riesigen Installation (55 Modalitäten, davon 10 CT-, 7 MR-, 8 CR- und 5 Thorax-Arbeitsplätze).

Unsere Ergebnisse zeigen, daß die Hochladekapazität für CR-Bilder am größten ist, während die pro Bild wesentlich kleineren CT- und MR-Bilder nur ein kleineres Volumen zulassen. Dies kann zur Erklärung der Beobachtung von

Clark et al. beitragen, der feststellte, daß in seiner Institution gegen Nachmittag die Latenzzeit für CT- und MR-Bilder deutlich anstieg, während sie für CRs konstant blieb [18]. Allerdings spielen bei den dort durchgeführten Messungen eine Reihe von Faktoren außerhalb des BVS eine Rolle und Clark et al. geben eine Reihe weiterer plausibler Gründe für dieses Resultat an. Unter anderem werden nach seinen Angaben vor allem nachmittags größere Serien zu Studienzwecken durchgeführt und frühere Serien werden zur Vorbereitung von Besprechungen an das BVS übertragen.

Es wird schnell deutlich, daß gleichzeitiges Herunterladen von Bildern die Hochladekapazität des BVS einschränkt. Auf der anderen Seite wurden die Messungen wie oben erwähnt mit 16 anfordernden Klienten durchgeführt [9], diese Zahl an Downloads kann praktisch nicht erwartet werden (Siehe 5.1 Grundsätzliches zum Studienaufbau, Seite 77). Es ist deshalb nicht davon auszugehen, daß in einem täglichen Arbeitsumfeld die Hochladekapazität durch das Abrufen von Bildern stark gesenkt wird. Dennoch muß der Effekt berücksichtigt werden, da gerade in Zeiten mit hohem Betrieb viele Bilder an das System übertragen und zugleich viele abgerufen werden.

Gemäß unseren Ergebnissen kann es insbesondere während der Kernarbeitszeiten mehrere Stunden dauern bis der Hochladevorgang und vor allem die Nachbearbeitungsphase abgeschlossen sind und Bilder zur Verfügung stehen. Dies kann zu Problemen in klinischen Abteilungen führen, in denen schneller Zugriff auf Bilder notwendig ist. Zu nennen wären hier Operationssäle, Schockraum, Ambulanzen und Intensivstationen. Um dieses Problem zu beheben, existieren mehrere Ansätze. Eine Möglichkeit wäre, diese Arbeitsplätze mit DICOM-Workstations auszustatten, die das direkte Anfordern der Bilder vom PACS erlauben. Eine Alternative dazu wäre die Installation mehrerer getrennter BVS, um auf diese Weise die Arbeitslast aufzuteilen. Dies hätte zudem Vorteile im Hinblick auf Hochverfügbarkeit des Systems, bedeutet aber einen höheren Wartungsaufwand. Zum Dritten kann man das System so konfigurieren, daß Bilder, die für diese Abteilungen bestimmt sind, mit hoher Priorität bearbeitet werden.

Die Hochladekapazität ist stark abhängig von der Prozessorleistung und dem Arbeitsspeicher des BVS-Servers. Eine praktisch lineare Korrelation ist

zwischen Standardserver und „Kleinem Server“ erkennbar. Es ist deshalb davon auszugehen, daß neuere, leistungsfähigere Hardware die Hochladekapazität weiter steigert. Auf der anderen Seite ist zu beobachten, daß die in der Radiologie durch Mehrzeilen-CTs und zunehmende Zahl an digitalen Modalitäten produzierte Datenmenge stark zunimmt [22, 63, 68], so daß dieser Effekt zunichte gemacht wird.

Eine Möglichkeit zur Steigerung der Hochladekapazität wäre die direkte Integration des BVS in das PACS, um so den zusätzlichen Übertragungsschritt und ggf. Overhead bei der Verwaltung von Bildern einzusparen. Allerdings spart dieses Vorgehen nicht die Zeit ein, die für die Umwandlung der DICOM-Bilder in das Wavelet-Format nötig ist. Eine weitere Möglichkeit besteht darin, mehrere Server mit der Umwandlung der Bilder zu betrauen (Siehe 5.7 Optimierung der Leistung und der Ausfallsicherheit, Seite 97).

## **5.6 Server-Tests mit mehreren synchronisierten Klienten**

### **5.6.1 Ohne Hochladen**

Bei den Messungen mit einer steigenden Anzahl gleichzeitig Bilder anfordernder Klienten ergab sich überraschend eine grundsätzlich lineare Korrelation zwischen Klienten-Anzahl und Anzeigezeit. Wir erwarteten, daß sich ab einer bestimmten Klienten-Anzahl ein exponentielles Verhältnis einstellen würde. Dieses Ergebnis erklären wir damit, daß die Leistungsfähigkeit der Hauptbestandteile des Servers, namentlich des Prozessors und des RAM bei 16 gleichzeitig Bilder anfordernden Klienten nicht annähernd erreicht war. Demnach wäre eine noch größere Zahl an Klienten erforderlich gewesen, um ein exponentielles Verhalten nachzuweisen. Selbst in einer großen Bildverteilungsinstallation ist jedoch nicht davon auszugehen, daß mehr als 16 Arbeitsplätze tatsächlich gleichzeitig Bilder anfordern. Da der Vorgang der Anforderung in der Regel nur 5 Sekunden benötigt, ist es eher wahrscheinlich, daß wesentlich kleinere Zahlen zutreffen (Siehe 5.1 Grundsätzliches zum Studienaufbau, Seite 77). Aus diesem Grund wurden keine Untersuchungen mit noch mehr Klienten durchgeführt. Die Nutzung noch leistungsfähigerer Hardware würde diesen praktisch linearen Zusammenhang höchstwahrscheinlich

scheinlich nicht verändern, es ist nur davon auszugehen, daß die Steigerung der Anzeigzeit pro weitem Klienten noch geringer ausfällt.

An dieser Stelle sei noch einmal darauf hingewiesen, daß ein Ressourcenverbrauch auf dem Server nur während des Übertragungsvorganges stattfindet. Der Übertragungsvorgang jedoch nimmt nur einen Teil der Anzeigzeit in Anspruch, wie die Vortests gezeigt haben (Siehe Abbildung 22, Seite 56).

In den Vortests wurde auch festgestellt, daß nicht synchronisierte Bildabrufe keinen meßbaren Einfluß auf die Anzeigzeit haben.

Die Verringerung der Anzeigzeit bei zwei Prozessoren im Vergleich zu einem im Server ist sichtbar, bringt aber nur eine halbe Sekunde. Es ist vor diesem Hintergrund fragwürdig, ob zwei Prozessoren im Verteilungssystem tatsächlich sinnvoll sind, insbesondere, da der Unterschied mit steigender Klientenzahl konstant bleibt. Da der Prozessor eine teure Komponente darstellt, halten wir es für gerechtfertigt, für die Verteilung ein Einprozessorsystem zu nutzen. Sollte es später nötig werden, kann ein weiterer Prozessor hinzugefügt werden. In diesem Fall sollte aber die Möglichkeit eines zweiten Servers in Betracht gezogen werden, da dieser bessere Ergebnisse und zudem höhere Verfügbarkeit des Systems gewährleisten kann.

Das Herabsetzen des Arbeitsspeichers von der Standardkonfiguration (1 GB) auf 512 MB oder 128 MB steigerte in jedem Fall die Anzeigzeiten, diese blieben jedoch in allen Fällen unter den als Grenzwert für die klinische Nutzung angesehenen 5 Sekunden. Aus diesem Grund können die Kosten für Arbeitsspeicher als entscheidendster Faktor angesehen werden.

Bei einer Netzwerkgeschwindigkeit von nur 10 Mbit/s gegenüber 100 oder 200 Mbit/s ergibt sich ein deutlicher Anstieg der Anzeigzeit bei allen Bildtypen. In diesem Fall stellt die Netzwerkanbindung das schwächste Glied in der Herunterlade- und Anzeigekette dar. Aus diesem Ergebnis läßt sich ableiten, daß ein BVS-Server auf jeden Fall mit 100 Mbit/s angebunden werden sollte. Allerdings bedeutet dies auch, daß das gesamte Netzwerk der entsprechenden Institution betrachtet werden muß. Bei unserer Infrastruktur mit einem Gigabit-Backbone hatten wir zu keinem Zeitpunkt den Eindruck, daß die Ergebnisse durch das Netzwerk beeinflußt waren. Bei langsameren Netzwerken ist davon

auszugehen, daß die einzelnen Meßergebnisse negativ beeinflußt werden, die generellen Aussagen und Schlüsse aber unbeeinflußt bleiben.

Beim Vergleich zwischen verlustbehafteter und verlustfreier Sekundärkompression zeigt sich, daß mit steigender Klienten-Anzahl die Nachteile von verlustbehafteter Sekundärkompression überwiegen. Da das Netzwerk mit verlustfreier Kompression keine Engstelle darstellte und bei verlustbehafteter Kompression die Datenmenge abnimmt, ist davon auszugehen, daß der Server selbst für die schlechten Ergebnisse der verlustbehafteten Sekundärkompression verantwortlich ist. Die Hardware der Klienten wurde so gewählt, daß sie sich nicht bremsend auswirkte [11]. Die Kompression der Bilder in Echtzeit vor dem Verschicken produziert offensichtlich eine derartige Last auf dem Server, daß dieser zum schwächsten Glied in der Herunterlade- und Anzeigekette wurde. Die generelle Nutzung der verlustbehafteten Sekundärkompression erscheint nach unseren Ergebnissen in keinem Fall angezeigt. Einzig bei der Übertragung von CR-Bildern über ein 10 Mbit/s-Netz brachte sie bessere Ergebnisse. In einem sehr langsamen Netzwerk kann sie zwar Vorteile bringen, jedoch fragt sich, ob ein solches Netz überhaupt für die Bildverteilung geeignet ist. An dieser Stelle sei darauf hingewiesen, daß das Exhibit-BVS die Möglichkeit der verlustbehafteten Primärkompression bietet, falls die vorhandene Netzwerk-Infrastruktur zu schlechten Ergebnissen führt. Diese sollte jedoch nur nach reiflicher Überlegung eingesetzt werden, da eine verlustfreie Darstellung dann nicht mehr möglich ist. Allerdings zeigt sich in neueren Publikationen, daß die verlustbehaftete Kompression keine klinisch relevanten Nachteile bietet [53, 61], die rechtliche Lage und die Akzeptanz beim Anwender oder Kunden ist jedoch ausschlaggebend.

Bis heute konnte keine detaillierte Literatur zum Thema Serverleistung bei webbasierten BVS gefunden werden. Die existierenden Zitate [5, 23, 57] sind recht ungenau, deshalb wird hier nicht weiter darauf eingegangen.

Zusammenfassend kann gesagt werden, daß eine Serverkonfiguration oberhalb eines Pentium III mit 700 MHz, 256 MB RAM und eine Netzwerk-anbindung mit mindestens 100 Mbit/s ausreicht, um auch mehrere Arbeitsplätze mit Bildern zu versorgen. Dabei bleiben die Anzeigezeiten unter 5 Sekunden. Verlustbehaftete Sekundärkompression sollte grundsätzlich nicht

verwendet werden, es sei denn, daß langsame Netzwerkverbindungen (10 Mbit/s) oder schwache Hardware auf Klientseite dies notwendig machen.

### **5.6.2 Mit Hochladen**

Die Ergebnisse zeigen einen deutlichen Einfluß gleichzeitigen Hochladens auf das Herunterladen von Bildern. Beim Vergleich zwischen Anzeigezeiten und Hochladekapazität zeigt sich erstaunlicherweise, daß eine höhere Hochladekapazität eines Bildtyps einen größeren negativen Einfluß auf die Anzeigezeiten mit sich bringt. Je mehr von einem Bildtyp pro Zeiteinheit hochgeladen werden kann, desto langsamer werden während des Hochladevorganges Bilder verteilt.

Der Standardserver war gegenüber dem „Kleinen Server“ in allen Fällen, in denen CRs hoch- oder heruntergeladen wurden überlegen. Da Projektionsbilder noch immer einen großen Teil aller Untersuchungen ausmachen, scheint die Auswahl eines größeren Servers angezeigt. Die Anwendung verlustbehafteter Primärkompression verbesserte beim „Kleinen Server“ die Ergebnisse für alle Bildtypen leicht, konnte aber die Vorteile des größeren Servers bei CR-Bildern nicht wettmachen.

Da bei 16 synchronisierten Klienten alle Anzeigezeiten über den als klinisch akzeptabel definierten 5 Sekunden lagen, wurde die Klient-Anzahl schrittweise reduziert, um die Zahl zu finden, bei der trotz gleichzeitigem Hochladen die Anzeigezeit unter 5 Sekunden bleibt. Außer beim Herunterladen von CT-Tableaus während dem Hochladen von CR-Bildern konnten vier bis sechs Klienten während dem Hochladen in akzeptabler Zeit mit Bildern versorgt werden. Auch diese gering scheinende Zahl stellt kein grundsätzliches Argument gegen webbasierte Bildverteilung dar, da vier bis sechs simultan bediente Klienten durchaus eine brauchbare Zahl darstellen [9] (Siehe 5.1 Grundsätzliches zum Studienaufbau, Seite 77). Zudem ist davon auszugehen, daß modernere Server oder andere Architekturen bessere Ergebnisse liefern. Auch die in 5.7 Optimierung der Leistung und der Ausfallsicherheit (Seite 97) beschriebene Aufteilung der Aufgaben auf mehrere Server schafft hier Abhilfe.

Unter Berücksichtigung aller Ergebnisse zeigt sich, daß erst bei gleichzeitigem Hoch- und Herunterladen von Bildmaterial die Grenzen unserer Server-

Hardware erreicht wurden. Um einschätzen zu können, welche Server-Hardware zum Einsatz kommen soll, muß zunächst analysiert werden, wie viele Bilder welchen Typs pro Zeiteinheit verarbeitet werden sollen. Dabei sind nicht nur die neu angefertigten Studien von Bedeutung, sondern eventuell auch Voraufnahmen, die zur Verfügung gestellt werden sollen. Erst unter Berücksichtigung der Ergebnisse mit gleichzeitigem Hoch- und Herunterladen zeigt sich, daß der Server unter keinen Umständen kleiner sein sollte, als der von uns verwendete Standardserver. Modernere Hardware mit zwei aktuellen Prozessoren und mindestens einem GB RAM dürfte allerdings Anzeigezeiten von unter 5 Sekunden für eine brauchbare Anzahl an Klienten sicherstellen.

Auf der Grundlage dieser Ergebnisse ist es dringend empfehlenswert, die Verteilung der Bilder und das Annehmen und Umwandeln der Bilder durch verschiedene Server durchführen zu lassen, wie in 5.7 Optimierung der Leistung und der Ausfallsicherheit (Seite 97) beschrieben. Ein einzelner Server kann nach unseren Ergebnissen problemlos eine große Zahl an Klienten mit Bildern versorgen. Das Annehmen und Komprimieren der Bilder hingegen sollte durch einen oder mehrere getrennte Server erfolgen. Dies stellt eine gleichbleibende akzeptable Anzeigezeit für die Anwender sicher und erlaubt zudem eine spätere Erweiterung der Hochladekapazität durch das Hinzufügen weiterer Annahme- und Umwandlungsserver.

Die Anwendung von verlustbehafteter Primärkompression erlaubt eine etwas höhere Hochladekapazität und eine geringe Verbesserung der Anzeigezeiten. In neueren Publikationen zeigt sich, daß verlustbehaftete Primärkompression keine klinisch relevanten Nachteile aufweist [53, 61]. Deshalb kann verlustbehaftete Primärkompression in Betracht gezogen werden, doch stehen rechtliche und organisatorische Faktoren sowie die Akzeptanz beim Anwender dem im Wege. Die Anwendung von verlustbehafteter Sekundärkompression ist nur dann ratsam, wenn nur eine schlechte Netzwerkinfrastruktur verfügbar ist (z. B. im Rahmen telemedizinischer Anwendungen) [9, 11] oder Bilder in einem WAN verteilt werden sollen. Auch während des Hochladens wurden Messungen mit verlustbehafteter Sekundärkompression durchgeführt. Die Ergebnisse bestätigen jedoch diese Aussagen und werden deshalb hier nicht weiter diskutiert.



## 5.7 Optimierung der Leistung und der Ausfallsicherheit

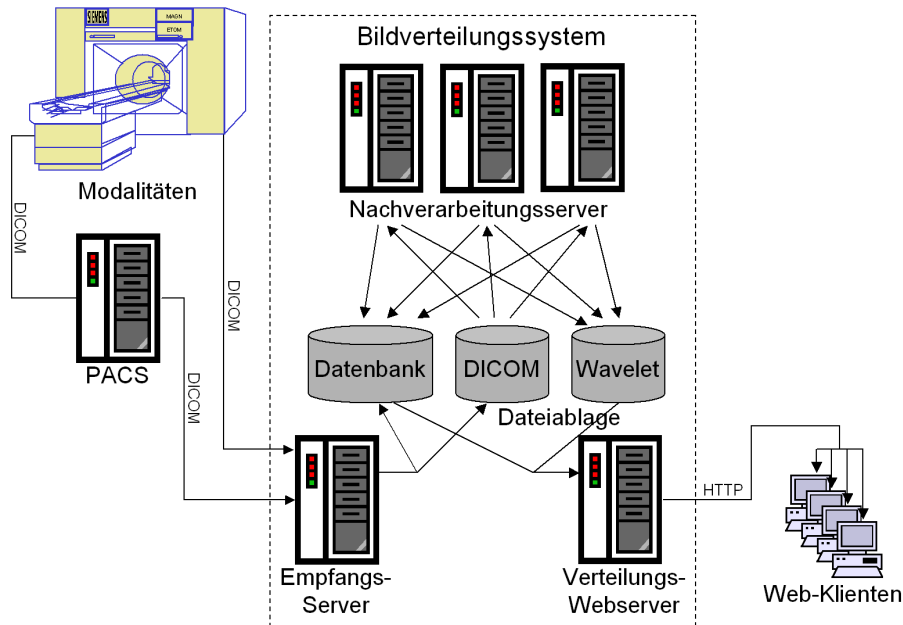
Der einfachste Weg, die Hochladekapazität und die Verteilungskapazität eines Bildverteilungssystems zu steigern, ist der Einsatz leistungsfähigerer Hardware. Wie bereits erwähnt, nimmt die Menge an in der Radiologie erzeugten Bilddaten durch den Einsatz von Mehrzeilen-CTs und zunehmend digitalen Modalitäten stark zu [63, 68]. Dies hat zur Folge, daß die durch ständige Hardwareentwicklung erreichten Leistungssteigerungen kompensiert werden. Der in anderen Bereichen übliche Ansatz, einfach auf schnellere Hardware zu warten, bietet deshalb keine Lösung.

Eine andere Möglichkeit ist der Einsatz einzelner Server für bestimmte Abteilungen. Dies hat jedoch eine Steigerung des Wartungsaufwandes zur Folge und sorgt, insbesondere bei den Zugriffsberechtigungen, für ein unübersichtliches Gesamtsystem [37]. Spätestens im Rahmen von Verlegungen müssen die Bilder transferiert werden, was den Umwandlungsprozeß neu anstößt und Zeit kostet. Der Wartungsaufwand und ggf. auch Lizenzkosten steigen mit der Einführung einer großen Zahl von abteilungseigenen Bildverteilungssystemen enorm an. Zudem ist es bei der Integration in ein Klinisches Arbeitsplatzsystem (KAS) hilfreich, nur ein BVS an die gesamte KAS-Installation anbinden zu müssen.

Eine weitere Möglichkeit zur Leistungssteigerung besteht darin, mehrere Server in intelligenter Weise am Bildverteilungsprozeß teilnehmen zu lassen, wobei diese auf eine gemeinsame Datenbasis zugreifen und somit gemeinsam ein großes Bildverteilungssystem darstellen.

Unsere Ergebnisse zeigen, daß zur Verteilung der Bilder ein einzelner Server ausreichend ist. Auch das Entgegennehmen der Bilder vom PACS bzw. den Modalitäten wurde von unserem Server gut bewältigt. Engpaß war die Umwandlung der Bilder in Wavelets und JPEGs, was wir als Nachbearbeitung bezeichnen. Daraus ergibt sich der Ansatz, mehrere Server, eine sogenannte Serverfarm, mit der Nachbearbeitung der Bilder zu betrauen. Die Architektur des vorliegenden BVS dürfte dies aufgrund der klaren Aufteilung zwischen SSCP-Prozeß, der die Bilder annimmt, und Curator-Prozeß, der sie umwandelt, ohne besonderen zusätzlichen Entwicklungsaufwand erlauben. Neue Bild-

verteilungssysteme sollten von Anfang an so entwickelt werden, daß eine Serverfarm die Nachbearbeitung übernehmen kann.



**Abbildung 40: Möglicher Aufbau eines webbasierten Bildverteilungssystems mit mehreren Servern, die jeweils bestimmte Aufgaben übernehmen. Zur Sicherstellung einer Hochverfügbarkeit des Systems können die einzelnen Server auch andere Aufgaben übernehmen.**

Die Ergebnisse, die ohne gleichzeitiges Hochladen beim Verteilen von Bildern erzielt wurden, belegen, daß eine derartige Architektur sehr erfolgreich wäre. Es zeigt sich, daß der mit der Verteilung der Bilder betraute Server keiner starken Belastung ausgesetzt ist, also vergleichsweise schwach ausgelegt sein kann, wenn er nicht gleichzeitig Bilder entgegennehmen oder umwandeln muß. Im Gegensatz dazu müssen die mit der Nachbearbeitung betrauten Rechner sehr leistungsfähig sein. Der Server, der mit der Entgegennahme der Bilder vom PACS oder von den Modalitäten betraut ist, sollte über eine schnelle Netzwerkanbindung verfügen, kann sich aber an der Nachbearbeitung beteiligen. Falls das webbasierte Bildverteilungssystem direkt in das PACS integriert wird, kann auf einen Server für die Entgegennahme der Bilder verzichtet werden.

Diese Architektur hätte den Vorteil, daß sich eine hohe Ausfallsicherheit einrichten ließe, indem ein anderer Server die Aufgabe des Empfangens oder Verteilens übernimmt, falls der primär mit dieser Aufgabe betraute Server

ausfällt. Der Ausfall eines einzelnen Umwandlungsservers würde ohnehin nur auf die Performanz und nicht auf die Verfügbarkeit des Systems Auswirkungen haben. Lediglich ein Ausfall der Datenbank oder der gemeinsamen Dateiablage würde zu Problemen führen, jedoch gibt es heute eine Reihe von etablierten Methoden, sowohl Datenbanken als auch Dateisysteme sehr ausfallsicher auszulegen.

Die gemeinsame Dateiablage muß an alle beteiligten Server mit ausreichender Geschwindigkeit angebunden sein. Dazu eignen sich Network Attached Storage- (NAS) oder auf Fibrechannel-Technologie basierende Storage Area Networks (SAN) sowie diverse Enterprise Storage Systeme. Es ist möglich, aber im Sinne der Hochverfügbarkeit nicht ratsam, auf die gemeinsame Dateiablage zu verzichten. Im einfachsten Fall kann einer der am BVS beteiligten Server über große Festplatten verfügen und die Daten über eine schnelle Netzwerkverbindung den übrigen Servern zur Verfügung stellen. In einem anderen Szenario würde der empfangende Server die Bilder direkt an einen bestimmten Umwandlungsserver weiterleiten. Nach der Umwandlung würden die Bilder direkt auf dem Verteilungsserver gespeichert. Dies führt aber dazu, daß die einzelnen Server nicht ohne weiteres untereinander Aufgaben austauschen können. Ein Ausfall eines Servers führt damit zu Verlust oder Nichtverfügbarkeit von Bilddaten oder sogar zum Ausfall des Gesamtsystems.

Ein weiterer wesentlicher Vorteil dieser Architektur liegt darin, daß sie gezielt erweitert werden kann, wenn sich Engstellen zeigen. Dies setzt natürlich ein entsprechendes Monitoring voraus, um diese Engstellen zielsicher ausmachen zu können. Steigt das von der Radiologie erzeugte Bilddatenvolumen an, so kann ein weiterer, leistungsfähiger Empfangsserver hinzugeführt und ggf. weitere Nachbearbeitungsserver nachgerüstet werden. Sollte es entgegen unseren Ergebnissen zu unzumutbaren Ladezeiten kommen, kann auch die Aufgabe der Bildverteilung auf mehrere Server verteilt werden.

## **6 Zusammenfassung**

### **6.1 Meßmethodik**

Unsere Ergebnisse zeigen, daß die auf dem Auslesen von Bildschirmpixeln basierende Methode 3 sehr stabil läuft und leicht zu interpretierende Resultate liefert. In dieser Hinsicht ist sie allen anderen Methoden überlegen. Der Zusatzaufwand der Softwareentwicklung machte sich bezahlt. Die auf Leistungsindikatoren (perfmon) basierende Methode 1 wird für Vortests empfohlen. Sie ist leicht zu implementieren, allerdings sind die Ergebnisse schwerer auszuwerten. Zusätzlich kann jedoch die Analyse von Leistungsindikatoren interessante Einblicke in die inneren Abläufe des BVS liefern. Phototransistoren (Methode 2) können angewendet werden, um jeden Einfluß auf das gemessene System auszuschließen.

### **6.2 Messungen**

Unsere Ergebnisse zeigen, daß bei der Einführung eines Bildverteilungssystems der pro Zeiteinheit auf den Server übertragenen Datenmenge besondere Beachtung geschenkt werden muß. Folgende Aussagen gehen von einer maximalen Anzeigzeit von 5 Sekunden aus, die für klinische Anwendungen als ausreichend erachtet wird.

1. Hochleistungsfähige Hardware ist auf dem Klient-PC nicht nötig.
2. Rechner, die älter als ein Pentium II mit 350 MHz sind, sollten nur für gelegentliches Nachschlagen verwendet werden. In diesem Fall sollte verlustbehaftete Sekundärkompression verwendet werden, die Auflösung sollte 1280 x 1024 Pixel nicht überschreiten.
3. Auf schnelleren PCs sollte verlustbehaftete Sekundärkompression nur dann verwendet werden, wenn die verfügbare Netzwerkbandbreite unterhalb von 10 Mbit/s liegt. Generell sind 128 MB Arbeitsspeicher oder mehr zu bevorzugen und eine Bildschirmauflösung über 1280 x 1024 kann verwendet werden.
4. Windows 2000 ist auf modernen Rechnern (>PII/350MHz) Windows NT überlegen, auf älteren Rechnern ist Windows NT schneller. Daraus ist

abzuleiten, daß zum Erzielen optimaler Ergebnisse auf moderneren Rechnern moderne Betriebssysteme eingesetzt werden sollten.

5. Die Hochladekapazität wird von simultanem Herunterladen geringfügig beeinflusst, hängt aber maßgeblich vom hochgeladenen Bildtyp und der Serverkonfiguration ab. Mit unserem Standardserver kann eine Hochladekapazität von 4 GB/h (96GB/t) erreicht werden. Dies scheint sogar für große Institutionen ausreichend.
6. Das gleichzeitige Hochladen von Bildern bremst die Anzeigeweiten deutlich. Während des Hochladens können nur vier bis sechs Klienten mit Anzeigeweiten von fünf Sekunden versorgt werden. Mit leistungsfähigerer Hardware scheint die Versorgung einer größeren Zahl von Klienten möglich.
7. Da Hochladekapazität und Anzeigeweite von einer stärkeren leistungsfähigeren Serverhardware deutlich profitieren, scheint ein Server mit zwei aktuellen Prozessoren und mindestens 1 GB RAM empfehlenswert.
8. Verlustbehaftete Primärkompression bringt gewisse Vorteile für Hochladekapazität und Anzeigeweite, wird aber nicht empfohlen, um die volle Bildqualität nicht zu verfälschen. Die Entscheidung für oder gegen verlustbehaftete Primärkompression ist jedoch von organisatorischen und rechtlichen Fragen überlagert.
9. Der Prozeß des Hochladens kann mehrere Stunden in Anspruch nehmen, was letztendlich klinisch inakzeptabel wird. Dies kann durch verbesserte Hardware oder eine veränderten BVS-Architektur oder mehrere intelligent zusammenarbeitende Server angegangen werden.
10. Webbasierte Bildverteilungssysteme sollten dezidierte Server für Empfang, Umwandlung und Verteilung der Bilder verwenden, um Hochverfügbarkeit, Performanz und spätere Erweiterbarkeit zu gewährleisten. Auf Empfangs-server kann verzichtet werden, wenn das BVS voll in das PACS integriert wird.

## **7 Summary**

### **7.1 Methods**

Our results show that method 3 based on reading the color of pixels was very stable and produced easily interpretable results. In that respect it is superior to all other methods. The initial work to write the self made software proved worth it. Both other methods remain a possibility to perform measurements. Method 1 which is based on performance counters (perfmon) can be recommended for pretests. It is simple yet stable, but interpretation of results can be difficult. The analysis of performance counters, however, may provide additional interesting insights into the internal workings of an application. Phototransistors (method 2) can be employed to eliminate any influence on the system to be evaluated.

### **7.2 Results of measurements**

Our results show that the amount of data transferred to the web server requires special attention. The following statements assume a maximum time to display (TTD) of 5 seconds, which is deemed clinically feasible.

1. High-end hardware is not required.
2. The usage of PCs older than Pentium II (350 MHz) can only be recommended for occasional image review purposes. Lossy compression should then be activated, the screen resolution should not exceed 1280 x 1024 pixels.
3. For PCs better than the above, lossy compression should only be used when the available network bandwidth is less than 10 Mbit/s. In general a RAM of 128 MB or more is preferable and screen resolutions above 1280 x 1024 pixels may be used when required.
4. Windows 2000 produced better results on more recent PCs (>PII/350MHz) than Windows NT, the opposite is true for older PCs. To archive optimal performance, recent operating systems should be employed on recent hardware.

5. The Upload Capacity (UC) was slightly influenced by a simultaneous download, but varied significantly for different image types. The server size also had a strong impact. An effective UC of 4.0 GB/h or 96 GB/d can be achieved, which appears appropriate even for large institutions.
6. On the other hand, a simultaneous upload significantly prolonged the TTD and only with up to 4-6 concurrent clients TTDs below 5 seconds could be achieved. The larger server performed superior and therefore a higher amount of concurrent clients seems possible with more recent server hardware.
7. Since both the UC as the TTD substantially benefited from the larger hardware, a standard server with at least two state-of-the-art processors and 1 GB of RAM appears recommendable.
8. Lossy primary compression demonstrated some benefit for the UC and the TTD but is not necessarily recommended in order to maintain the original character of the images.
9. The upload process may take hours and hence delay the availability of images for display to a clinically unacceptable point. This delay can potentially be reduced by improvements of the server hardware but would preferably be addressed by a modified IWS architecture or multiple interconnected IWS servers.
10. Web based Image Distribution Systems should employ dedicated servers for reception, transformation and distribution of images to ensure high availability, performance and future upgrades. A server dedicated to reception can be omitted, if the IWS is integrated into the PACS.

## 8 Literaturverzeichnis

- 1 Achenbach S (2002) Intranet and radiology: a critical appraisal of radiological applications of Intranet technology. Eur Radiol 12(2):485-90
- 2 Bandon D, Ligier Y, Trayser G, Girard C, Logean M, Ratib O (1998) A hospital-wide distributed PACS based on intranet. Medinfo 9 Pt 2:1075-1079
- 3 Bellon E, Feron M, Deprez T, Pauwels H, Vanautgaerden M, De Deurwaerder A, Reynders R, Reviere W, Draelants B, Suetens P, Marchal G, Van Den Bosch B (2002) Integrating images into a central medical information system. Stud Health Technol Inform 93:53-60
- 4 Bellon E, Wauters J, Fernandez-Bayo J, Feron M, Verstreken K, Van Cleynenbreugel J, Van den Bosch B, Desmaret M, Marchal G, Suetens P (1997) Using WWW and JAVA for image access and interactive viewing in an integrated PACS. Med Inform 22(4):291-300
- 5 Bennett WF, Spigos DG, Tzalonikou MT, Terrell JE, Augustyn MA (1999) Web-based viewing of picture archiving and communications systems images--Part I: Optimal personal computer configuration. J Digit Imaging 12(2 Suppl 1):112-115
- 6 Bennett WF, Spigos DG, Tzalonikou MT, Terrell JE, Augustyn MA (1999) Web-based viewing of picture archiving and communications systems images-Part II: The effect of compression on speed of transmission. J Digit Imaging May; 12(2 Suppl 1):116-118
- 7 Bennett WF, Tunstall KM, Skinner PW, Spigos DG (2002) Delivering images to the operating room: a web-based solution. J Digit Imaging 15 Suppl 1:137-139
- 8 Bergh B, Felix R (1999) WWW-based Access to Radiological Patient Data: Two Years of Experience. In: Nehrlich M, Kretschmer R (ed) *Studies in Health Technology and Informatics: The Impact of Telemedicine on Health Care Management*. IOS Press, Amsterdam Berlin Oxford Tokyo Washington, pp 217-229
- 9 Bergh B, Pietsch M, Schlaefke A, Garcia I, Vogl TJ (2003) Performance of web-based image distribution: server oriented measurements. Eur Radiol Nov; 13(11):2419-24
- 10 Bergh B, Pietsch M, Schlaefke A, Garcia I, Vogl TJ (2004) Upload capacity and time-to-display of an image Web system during simultaneous up- and download processes. Eur Radiol Mar; 14(3):526-33



- 11 Bergh B, Pietsch M, Schlaefke A, Vogl TJ (2003) Performance of web-based image distribution: client oriented measurements. Eur Radiol Sep; 13(9):2161-9
- 12 Bergh B, Pietsch M, Vogl TJ (2000) Internet technology for clinical applications in a digital radiography department. Radiologe 40:371-38
- 13 Bick U, Lenzen H (1999) PACS: the silent revolution. Eur Radiol 9(6):1152-1160
- 14 Blackmore CC, Richardson ML, Linnau KF, Schwed AM, Lomoschitz FM, Escobedo EM, Hunter JC, Jurkovich GJ, Cummings P (2003) Web-based image review and data acquisition for multiinstitutional research. AJR Am J Roentgenol May; 180(5):1243-6.
- 15 Boehm T, Handgraetinger O, Link J, Ploner R, Voellmy DR, Marincek B, Wildermuth S (2004) Evaluation of radiological workstations and web-browser-based image distribution clients for a PACS project in hands-on workshops. Eur Radiol May; 14(5):908-14.
- 16 Boochever SS (2004) HIS/RIS/PACS integration: getting to the gold standard. Radiol Manage May-Jun; 26(3):16-24; quiz 25-7.
- 17 Browning GC, Liang Y, Buckwalter KA, Kruger RA, Aisen A (1996) World Wide Web interface to digital imaging and communication in medicine-capable image servers. J Digit Imaging 9(4):178-84
- 18 Clark KW, Melson DL, Moore SM, James Blaine GJ, Moulton RA, Clayton WK, Peterson CS, Vendt BA (2003) Tools for managing image flow in the modality to clinical-image-review chain. J Digit Imaging Sep; 16(3):310-7.
- 19 de Azevedo-Marques PM, Carita EC, Benedicto AA, Sanches PR (2004) Integrating RIS/PACS: the web-based solution at University Hospital of Ribeirao Preto, Brazil. J Digit Imaging Sep; 17(3):226-33.
- 20 de Regt D, Weinberger E (2004) MyFreePACS: a free web-based radiology image storage and viewing tool. AJR Am J Roentgenol Aug; 183(2):535-7.
- 21 Doyle AJ, Gunn ML, Gamble GD, Zhang M (2002) Personal computer-based PACS display system: comparison with a dedicated PACS workstation for review of computed radiographic images in rheumatoid arthritis. Acad Radiol Jun; 9(6):646-53.
- 22 Dumery B (2002) Digital image archiving: challenges and choices. Radiol Manage May-Jun; 24(3):30-8; quiz 39-41.

- 23 Fernandez-Bayo J, Barbero O, Rubies C, Sentis M, Donoso L (2000) Distributing medical images with internet technologies: a DICOM web server and a DICOM java viewer. Radiographics 20(2):581-590
- 24 Foord K (2001) Year 2000: status of picture archiving and digital imaging in European hospitals. Eur Radiol 11(3):513-524
- 25 Foord KD (1999) PACS workstation respecification: display, data flow, system integration, and environmental issues, derived from analysis of the Conquest Hospital pre-DICOM PACS experience. Eur Radiol 9(6):1161-1169
- 26 Fründ R, Techert J, Strotzer M, Börner W, Tsakpinis A, Feuerbach S (2001) Das PACS-Konzept der Universität Regensburg. RöfoFortschr Geb Rontgenstr Neuen Bildgeb Verfahr 173(4):362-367
- 27 Gay SB, Sobel AH, Young LQ, Dwyer SJ 3rd (2002) Process involved in reading imaging studies: workflow analysis and implications for workstation development. J Digit Imaging Sep; 15(3):171-7; discussion 170.
- 28 Gropper A, Doyle S, Dreyer K (1998) Enterprise-scale image distribution with a Web PACS. J Digit Imaging 11(3 Suppl 1):12-7
- 29 Harrison SW (2003) Success with Web-based image access. Radiol Manage Mar-Apr; 25(2):36-8.
- 30 Henri CJ, Rubin RK, Cox RD, Bret PM (1997) Design and implementation of World Wide Web-based tools for image management in computed tomography, magnetic resonance imaging, and ultrasonography. J Digit Imaging 10(3 Suppl 1):77-9
- 31 Himes JA, Rosenfeld K (2004) Integrating digital systems: commitment and collaboration. Radiol Manage May-Jun; 26(3):28-30.
- 32 Jost RG, Blaine GJ, Fritz K, Blume H, Sadhra S (2002) Enhanced interfaces for web-based enterprise-wide image distribution. J Digit Imaging 15 Suppl 1:189-93.
- 33 Kalthoff, O Lecture JPEG Basics. <http://www.mi.fh-heilbronn.de/>
- 34 Kalthoff, O Lecture Notes on Imaging Physics. <http://www.mi.fh-heilbronn.de/>
- 35 Kalyanpur A, Weinberg J, Neklesa V, Brink JA, Forman HP (2003) Emergency radiology coverage: technical and clinical feasibility of an international teleradiology model. Emerg Radiol Dec; 10(3):115-8. Epub 2003 Jul 22.

- 36 Knoll P, Holl K, Mirzaei S, Koriska K, Kohn H (2000) Distributed nuclear medicine applications using World Wide Web and Java technology. Eur Radiol 10(9):1483-6
- 37 Kotter E, Jager D, Binder M, Roesner A, Zaiss A, Pelikan E, Ghanem N, Winterer J, Uhrmeister P, Klar R, Timmermann U, Langer M (2003) Elektronische Befund- und Bildverteilung aus einem PACS: Umsetzung der datenschutzrechtlichen Aspekte am Beispiel des Universitätsklinikums Freiburg. RöfoJun; 175(6):849-54.
- 38 Kotter E, Langer M (1998) Integrating HIS-RIS-PACS: the Freiburg experience. Eur Radiol 8(9):1707-1718
- 39 Lee JS, Tsai CT, Pen CH, Lu HC (2003) A real time collaboration system for teleradiology consultation. Int J Med Inform Dec; 72(1-3):73-9.
- 40 Luccichenti G, Ngo Dinh N, Cademartiri F, Evangelisti G, Paolillo A, Bastianello S. (2004) Teleradiology system accessible through a common web browser. Radiol Med (Torino) Nov-Dec; 108(5-6):542-8.
- 41 Maldjian JA, Liu WC, Hirschorn D, Murthy R, Semanczuk W (1997) Wavelet transform-based image compression for transmission of MR data. Am J Roentgenol 169(1):23-6
- 42 Mallat S (1989) A Theory for Multiresolution Signal Decomposition: The Wavelet Representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 2(7)
- 43 Mascarini C, Ratib O, Trayser G, Ligier Y, Appel RD (1996) In-house access to PACS images and related data through World Wide Web. Eur J Radiol 22(3):218-20
- 44 McEney KW, Suitor CT, Thompson SK, Shepard JS, Murphy WA. (2002) Evaluation of soft copy distribution of diagnostic studies before soft copy interpretation. J Digit Imaging 15 Suppl 1:76-80.
- 45 Melson DL, Moore SM, Blaine JG, Clark KW, Sadhra S, Blume H (2002) Challenges in image acquisition and distribution for clinical image service. J Digit Imaging 15, Suppl 1:144-150
- 46 Mildenerger P, Eichelberg M, Martin E (2002) Introduction to the DICOM standard. Eur Radiol 12(4):920-927
- 47 Mildenerger P, Jensch P (1999) Verwendung des DICOM-Standards in heterogener Umgebung Inkompatibilität oder Interoperabilität? Radiologe 1999 Apr; 39(4):282-5
- 48 Morishita J, Watanabe H, Katsuragawa S, Oda N, Sukenobu Y, Okazaki H, Nakata H, Doi K (2005) Investigation of misfiled cases

- in the PACS environment and a solution to prevent filing errors for chest radiographs. Acad Radiol Jan; 12(1):97-103.
- 49 Mulvaney J (2002) The case for RIS/PACS integration. Radiol Manage May-Jun; 24(3):24-9.
- 50 Munch H, Engelmann U, Schroter A, Meinzer HP (2004) The integration of medical images with the electronic patient record and their web-based distribution. Acad Radiol Jun; 11(6):661-8.
- 51 Nissen-Meyer S, Holzknrecht N, Wieser B, Francke M, Küttner B, Adelhard K, Villain S, Weber J, Reiser M. (2002) Produktivitätsverbesserung durch klinikweite RIS und PACS: eine Fallstudie. Radiologe May; 42(5):351-60.
- 52 Ozsunar Y, Keceli M, Koseoglu K, Coskun G, Karaman C (2003) PACS utilization in radiologic research. Acad Radiol Jan; 10(1):32-6.
- 53 Parisi SB, Mogel GT, Dominguez R, Dao H, Cramer TJ (1998) The effect of 10:1 compression and soft copy interpretation on the chest radiographs of premature neonates with reference to their possible application in teleradiology. Eur Radiol 8(1):141-143
- 54 Park JM, Ruess L, O'Connor SC, Hussain F, Oshiro DY, Person DA. (2004) Internet consultations from a remote Pacific island: impact of digitized radiologic images on referral decisions. J Digit Imaging Dec; 17(4):253-7.
- 55 Pennebaker, William B., JPEG still image data compression standard, Chapman&Hall, 1992
- 56 Ralston MD, Coleman R, Scrutchfield KM (2002) An organized pilot program promoting web access to picture archiving and communications system data: emphasis on offices outside the host institution. J Digit Imaging 15 Suppl 1:162-5. Epub 2002 Mar 20.
- 57 Ratib O, Ligier Y, Bandon D, Valentino D (2000) Update on digital image management and PACS. Abdom Imaging 25(4):333-340
- 58 Rosset A, Ratib O, Geissbuhler A, Vallee JP (2002) Integration of a multimedia teaching and reference database in a PACS environment. Radiographics Nov-Dec; 22(6):1567-77.
- 59 Schomer DF, Elekes AA, Hazle JD, Huffman JC, Thompson SK, Chui CK, Murphy WA (1998) Introduction to wavelet-based compression of medical images. Radiographics 18(2):469-481
- 60 Sterling L, Tait GA, Edmonds JF (2003) Interpretation of digital radiographs by pediatric critical care physicians using Web-based bedside personal computers versus diagnostic workstations. Pediatr Crit Care Med Jan; 4(1):26-32.

- 61 Sung MM, Kim HJ, Yoo SK, Choi BW, Nam JE, Kim HS, Lee JH, Yoo HS (2002) Clinical Evaluation of Compression Ratios using JPEG2000 on Computed Radiography Chest Images. J Digit Imaging 15(2):78-83
- 62 Szot A, Jacobson FL, Munn S, Jazayeri D, Nardell E, Harrison D, Drosten R, Ohno-Machado L, Smeaton LM, Fraser HS (2004) Diagnostic accuracy of chest X-rays acquired using a digital camera for low-cost teleradiology. Int J Med Inform 2004 Feb; 73(1):65-73.
- 63 Tamm EP, Thompson S, Venable SL, McEnery K (2002) Impact of multislice CT on PACS resources. J Digit Imaging 15 Suppl 1:96-101
- 64 Teichgraber UK, Lehmkuhl L, Harderer A, Emmel D, Ehrenstein T, Ricke J, Felix R (2003) Aufbau und Datentransferanalyse eines proprietären digitalen Open-Source-Bildarchivs für radiologische Dienstleistungen. Röfo Jul; 175(7):973-80.
- 65 Terrier F (2000) Web and PACS: heralding the new age of imaging in the health care community. Abdom Imaging 25(4):331
- 66 Treitl M, Wirth S, Lucke A, Nissen-Meyer S, Villain S, Trumm C, Rieger J, Pfeifer KJ, Reiser M. (2005) Möglichkeiten der Optimierung des Workflows in radiologischen Abteilungen jenseits von RIS und PACS Radiologe Jun 15 [Online first]
- 67 Trumm C, Dugas M, Wirth S, Treitl M, Lucke A, Kuttner B, Pander E, Clevert DA, Glaser C, Reiser M. (2005), Digitales Lehrarchiv Konzept, Implementation und Erfahrungen im universitären Umfeld, Radiologe Jun 22 [Online first]
- 68 van Elzakker T (2004) Current perspectives on CT technology. Radiol Manage Sep-Oct; 26(5):54-8.
- 69 Vogl R. (2005) Teleradiologie und PACS — das Konzept des Universitätsklinikums Innsbruck Radiologe Jun 11 [Online first]
- 70 Wang X, Hu J, Cao J, Wang B, Jiao J, Wei Y, Wang X, Luo M, Luo S. (2003) Set-up and preliminary performance of a digital radiology conference system. Chin Med J (Engl) May; 116(5):658-60.
- 71 Wendt G, Kulbago T, Peppler W (2000) Implementing a Java-based image and report distribution system in a non-picture archiving and communication system environment. J Digit Imaging 13(2 Suppl 1):142-4
- 72 Wirth S, Treitl M, Lucke A, Mittermaier I, Nissen-Meyer S, Villain S, Pfeifer KJ, Reiser M (2005) Empfehlungen zur Medienwahl für die Archivierung radiologischer Bilddaten auf der Basis einer Analyse der Zugriffszeiten auf verschiedene PACS-Archivebenen. Röfo Feb; 177(2):250-7.

- 73 Wirth S, Treitl M, Villain S, Lucke A, Nissen-Meyer S, Mittermaier I, Pfeifer KJ, Reiser M (2005) PACS: Speicherung und Abruf digitaler radiologischer Bilddaten. Radiologe Jun 8 [Online first]
- 74 Worthy S, Rounds KC, Soloway CB (2003) Strengthening your ties to referring physicians through RIS/PACS integration. Radiol Manage Mar-Apr; 25(2):18-22.
- 75 Yam CS, Levine D, Nishino M, Sitek A, Larson M (2005) A simple method for displaying cine images on web-based teaching files. AJR Am J Roentgenol Feb; 184(2):691-4.
- 76 Zhang J, Sun J, Stahl JN. (2003) PACS and Web-based image distribution and display. Comput Med Imaging Graph 27(2-3):197-206.

## 9 Tabellen

### 9.1 Entwicklung der Meßmethodik

Sekunden		verlustfrei			verlustbehaftet			
		CR	CT	MR	CR	CT	MR	
8	synchrone	gleiches Bild	1,74	3,32	2,51	2,78	3,56	7,60
	Klienten	versch. Bilder	1,80	3,36	2,54	2,10	3,53	5,20
16	synchrone	gleiches Bild	2,92	4,83	3,77	4,14	5,18	14,65
	Klienten	versch. Bilder	2,88	4,69	3,72	4,20	4,99	11,42

**Anzeigzeiten (in Sekunden) abhängig davon, ob von den 8 oder 16 synchronisierten**

**Klienten gleiche oder verschiedene Bilder angefordert wurden.**

Minuten	nacheinander	gleichzeitig
CR ? timp	29	
CT ? timp	28	
MR ? timp	71	
CR ? tpp	36	
CT ? tpp	73	
MR ? tpp	31	
alle ? timp		111
alle ? tpp		156
	<b>268</b>	<b>268</b>

**Ergebnisse des zweiten Vortests: Hochladezeit für alle Bilder der drei Sender-PCs.**

### 9.2 Klient-Hardware-Tests

Sekunden	Anmeldung	CT1 Miniaturans.	MR1 Miniaturans.
PI 90	41,22	2,39	4,04
PI 166	25,89	1,31	2,60
PI 233	22,11	1,07	2,08
PII 350	16,13	1,58	1,68
PIII 500	14,40	0,57	1,38
PIII 700	13,33	0,53	1,41
PIII 933	12,68	0,52	1,32

**Nicht-Bildanzeigepoperationen bei verschiedenen Prozessorgenerationen und Taktfrequenzen bei einer Auflösung von 1024 x 768 x 32.**

Sekunden	verlustfrei			verlustbehaftet		
	CR	CT	MR	CR	CT	MR
PI 90	11,21	25,44	18,07	9,11	25,65	14,75
PI 166	5,62	12,58	8,80	4,73	12,60	7,35
PII 350	2,17	4,77	3,55	1,94	4,77	3,31
PIII 500	1,69	3,68	2,81	1,51	3,71	2,79
PIII 700	1,33	3,22	2,51	1,23	3,22	2,48
PIII 933	1,12	2,60	2,07	1,03	2,62	2,22

**Anzeigzeiten (in Sekunden) für verschiedene Prozessoren bei einer Auflösung von 1024 x 768 x 32.**

Sekunden	MB	verlustfrei			verlustbehaftet		
		CR	CT	MR	CR	CT	MR
PI 166	64	5,31	11,99	8,47	4,57	12,05	7,30
	128	5,62	12,58	8,80	4,73	12,60	7,35
PII 350	64	2,25	5,06	4,02	1,96	5,01	3,93
	128	2,15	4,81	3,55	1,88	4,83	3,36
	256	2,12	4,68	3,48	1,88	4,68	3,27
PIII 700	64	1,59	3,63	3,23	1,37	3,43	3,56
	128	1,33	3,22	2,51	1,23	3,22	2,48
	256	1,41	3,10	2,46	1,22	3,12	2,50

**Anzeigezeiten (in Sekunden) für verschiedene Arbeitsspeichergößen. Die drei PCs entsprechen den drei Referenz-PCs.**

Sekunden	CR	LossyCR	MR	LossyMR	CT	LossyCT
1024x768x8	1,09	0,97	1,70	1,90	2,52	2,53
1024x768x16	0,95	0,88	1,67	1,88	2,49	2,50
1024x768x32	1,12	1,02	1,91	2,04	2,74	2,76
1152x864x8	1,26	1,12	1,78	1,94	2,61	2,63
1152x864x16	1,11	0,98	1,84	2,01	2,63	2,64
1152x864x32	1,24	1,10	2,04	2,18	2,79	2,81
1280x1024x8	1,41	1,27	1,82	2,00	2,62	2,64
1280x1024x16	1,33	1,20	1,96	2,08	2,76	2,76
1280x1024x32	1,54	1,36	2,26	2,33	3,04	3,04
1600x1200x8	2,83	2,33	2,04	2,19	2,91	2,92
1600x1200x16	2,61	2,20	2,12	2,26	2,95	2,95
1600x1200x32	3,28	2,76	2,69	2,61	3,46	3,47

**Anzeigezeiten (in Sekunden) für verschiedene Bildschirmauflösungen (angegeben in Pixeln). Die gezeigten Ergebnisse wurden mit dem Pentium III mit 700 MHz, einer AGP-Grafikkarte mit 8 MB RAM gemessen.**

Sekunden			CR	LossyCR	MR	LossyMR	CT	LossyCT
PI 166	10 MBit/s	Half	5,71	4,91	8,97	7,50	12,90	12,92
		Full	5,67	4,86	8,90	7,51	12,69	12,63
	100 MBit/s	Half	5,54	4,74	8,75	7,38	12,55	12,61
		Full	5,62	4,73	8,80	7,35	12,58	12,60
PII 350	10 MBit/s	Half	2,30	1,89	3,91	3,33	5,29	5,28
		Full	2,32	1,87	3,85	3,35	5,24	5,26
	100 MBit/s	Half	2,15	1,90	3,56	3,34	4,80	4,82
		Full	2,15	1,88	3,55	3,36	4,81	4,83
PIII 700	10 MBit/s	Half	1,71	1,26	2,81	2,46	3,62	3,63
		Full	1,66	1,27	2,74	2,50	3,59	3,57
	100 MBit/s	Half	1,41	1,20	2,46	2,51	3,14	3,16
		Full	1,33	1,23	2,51	2,48	3,22	3,22

**Anzeigezeiten (in Sekunden) bei verschiedenen Netzwerkgeschwindigkeiten (Mbit/s) gemessen mit den drei Referenz-PCs.**



Sekunden			CR	LossyCR	MR	LossyMR	CT	LossyCT
64 MB	PI 166	NT 4	5,31	4,57	8,47	7,30	11,99	12,05
		W2K	5,68	4,86	9,71	8,03	13,20	13,23
	PII 350	NT 4	2,25	1,96	4,02	3,93	5,06	5,01
		W2K	2,23	1,99	4,48	4,26	5,17	5,14
	PIII 700	NT 4	1,59	1,37	3,23	3,56	3,63	3,43
		W2K	1,32	1,13	2,74	2,86	3,32	3,22
128 MB	PI 166	NT 4	5,62	4,73	8,80	7,35	12,58	12,60
		W2K	6,04	5,23	9,55	8,08	13,69	13,74
	PII 350	NT 4	2,15	1,88	3,55	3,36	4,81	4,83
		W2K	2,10	1,82	3,47	3,25	4,89	4,87
	PIII 700	NT 4	1,33	1,23	2,51	2,48	3,22	3,22
		W2K	1,24	1,06	2,15	2,27	3,06	3,08

Anzeigzeiten (in Sekunden) bei verschiedenen Betriebssystemen mit den drei Referenz-PCs. NT4: Windows NT 4.0 Workstation; W2K: Windows 2000 Professional.

## 9.3 Server-Tests mit mehreren synchronisierten Klienten

### 9.3.1 Ohne Hochladen

Sekunden	Zwei Prozessoren						Ein Prozessor					
	verlustfrei			verlustbehaftet			verlustfrei			verlustbehaftet		
Klienten	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR
1	0,85	2,40	1,63	0,82	2,40	1,78	0,85	2,43	1,66	0,83	2,45	1,74
2	0,95	2,47	1,69	1,32	2,49	2,54	1,02	2,59	1,85	1,25	2,57	2,32
4	1,20	2,75	1,94	1,89	2,84	4,22	1,47	2,84	2,14	1,94	2,88	3,31
6	1,40	3,03	2,28	1,87	3,18	5,74	1,79	3,06	2,43	2,65	3,17	4,76
8	1,74	3,32	2,51	2,78	3,56	7,60	2,19	3,39	2,72	3,20	3,46	5,67
10	2,14	3,67	2,81	3,06	3,95	9,63	2,30	3,78	3,09	3,73	3,86	7,41
12	2,30	4,15	3,22	3,55	4,41	11,16	2,79	4,10	3,40	3,58	4,23	9,21
14	2,91	4,46	3,49	3,51	4,70	12,28	3,19	4,59	3,91	5,21	4,83	10,12
16	2,92	4,83	3,77	4,14	5,18	14,65	2,77	5,01	4,26	5,48	5,22	11,14

Anzeigzeiten (in Sekunden) für verschiedene Server-Prozessorzahlen und synchronisierte Klienten.

Sekunden	1 GB RAM						512 MB RAM						128 MB RAM					
	verlustfrei			verlustbehaftet			verlustfrei			verlustbehaftet			verlustfrei			verlustbehaftet		
Klienten	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR
1	0,85	2,40	1,63	0,82	2,40	1,78	0,86	2,43	1,64	0,82	2,38	1,80	0,89	2,52	1,72	0,91	2,50	1,77
2	0,95	2,47	1,69	1,32	2,49	2,54	1,32	2,47	1,72	0,97	2,51	2,58	0,94	2,53	1,77	1,03	2,58	2,66
4	1,20	2,75	1,94	1,89	2,84	4,22	1,25	2,73	1,98	1,45	2,84	4,26						
6	1,40	3,03	2,28	1,87	3,18	5,74	2,18	3,02	2,25	2,53	3,18	5,94						
8	1,74	3,32	2,51	2,78	3,56	7,60	2,46	3,45	2,55	2,90	3,58	7,99	2,68	3,50	2,51	2,25	3,67	13,54
10	2,14	3,67	2,81	3,06	3,95	9,63	2,59	3,77	2,87	2,97	3,99	10,03						
12	2,30	4,15	3,22	3,55	4,41	11,16	2,88	4,21	3,24	4,10	4,45	11,63						
14	2,91	4,46	3,49	3,51	4,70	12,28	3,04	4,62	3,59	4,39	4,88	13,47						
16	2,92	4,83	3,77	4,14	5,18	14,65	3,19	5,08	3,99	4,98	5,41	15,58	3,97	4,95	3,81	6,15	5,29	22,54

Anzeigzeiten (in Sekunden) für verschiedene Server-Arbeitspeicher-Konfigurationen und synchronisierte Klienten.

Sekunden	FastEtherChannel						100Mbit/s						10Mbit/s					
	verlustfrei			verlust-behaftet			verlustfrei			verlust-behaftet			verlustfrei			verlust-behaftet		
Klienten	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR	CR	CT	MR
1	0,85	2,40	1,63	0,82	2,40	1,78	0,86	2,43	1,66	0,82	2,40	1,79	1,20	2,94	2,02	0,88	2,91	1,74
2	0,95	2,47	1,69	1,32	2,49	2,54	0,96	2,46	1,71	0,98	2,52	2,55	2,08	3,60	2,46	1,10	3,60	2,52
4	1,20	2,75	1,94	1,89	2,84	4,22	1,31	2,77	1,99	1,48	2,85	4,15	2,93	4,73	3,10	2,14	4,75	4,10
6	1,40	3,03	2,28	1,87	3,18	5,74	1,64	3,04	2,26	1,72	3,18	5,72	3,69	5,85	3,81	2,23	5,95	5,74
8	1,74	3,32	2,51	2,78	3,56	7,60	1,78	3,40	2,52	2,76	3,55	7,56	4,57	7,09	4,61	2,67	7,10	7,34
10	2,14	3,67	2,81	3,06	3,95	9,63	1,94	3,85	2,82	3,32	3,98	9,54	5,21	8,43	5,59	3,31	8,45	9,57
12	2,30	4,15	3,22	3,55	4,41	11,16	2,44	4,23	3,17	3,44	4,51	11,06	5,91	9,94	6,58	3,59	10,08	10,96
14	2,91	4,46	3,49	3,51	4,70	12,28	2,52	4,64	3,47	3,53	4,91	12,82	6,67	11,56	7,94	3,96	11,61	12,81
16	2,92	4,83	3,77	4,14	5,18	14,65	3,29	5,08	3,90	3,85	5,39	15,07	8,24	12,99	8,65	5,98	13,20	14,57

**Anzeigezeiten (in Sekunden) für verschiedene Server-Netzwerkanbindungen und synchronisierte Klienten.**

### 9.3.2 Mit Hochladen

Sekunden			kein	Hochladen von ...			Nachverarbeitung von ...		
				MR	CT	CR	MR	CT	CR
TTD CR 16 Klienten		Standardserver	3,04	6,69	7,12	11,73	4,25	4,11	3,97
		Kleiner Server	3,34	10,19	13,72	22,85	4,21	4,06	4,14
		Kl. Server, verlustbeh.Primärkomp.	2,85	9,77	12,41	20,35	3,66	3,69	4,04
TTD CT 16 Klienten		Standardserver	5,18	8,20	9,71	12,72	5,79	5,73	5,36
		Kleiner Server	5,58	8,51	11,05	19,71	6,00	5,97	5,92
		Kl. Server, verlustbeh.Primärkomp.	4,83	7,36	8,43	20,92	5,02	4,92	4,94
TTD MR 16 Klienten		Standardserver	4,04	7,74	10,92	8,68	4,32	4,46	4,17
		Kleiner Server	4,37	7,17	9,63	17,10	4,65	4,74	4,84
		Kl. Server, verlustbeh.Primärkomp.	4,16	6,03	7,65	16,81	4,37	4,19	4,25

**Anzeigezeiten (TTD) (in Sekunden) für drei verschiedene Bildtypen mit 16 synchronisierten Klienten. Messungen wurden ohne Hochladen und während des Hochladens verschiedener Bildtypen durchgeführt. kein = kein Hochladen.**

Klienten	kein	Hochladen von ...			kein	Hochladen von ...			kein	Hochladen von ...		
		CT	CR	MR		CT	CR	MR		CT	CR	MR
1	0,86			1,23	2,66			3,07	1,76			2,26
2	1,12		3,63	2,48	2,74		6,21	3,79	1,85		3,84	2,45
4	1,47	3,18	4,22	2,81	3,19	4,69	6,93	4,21	2,18	3,48	3,79	3,14
8	1,95	3,34	6,36	3,30	3,57	6,72	10,21	4,91	2,64	5,29	5,30	3,78
12	2,48	5,23	9,05	4,18	4,29	8,21	11,46	6,53	3,26	8,10	6,99	5,33
16	3,04	7,12	11,73	6,69	5,18	9,71	12,72	8,20	4,04	10,92	8,68	7,74

**Anzeigezeiten (in Sekunden) für drei verschiedene Bildtypen mit verschieden vielen synchronisierten Klienten. Messungen wurden ohne Hochladen und während des Hochladens verschiedener Bildtypen durchgeführt. Nur der Standardserver mit verlustfreier Primärkompression wurde verwendet. kein = kein Hochladen.**

# 10 Verzeichnisse

## 10.1 Abkürzungen

ADC	Analog Digital Converter
AGP	Accelerated Graphics Port
API	Application Programming Interface
BVS	Bildverteilungssystem
CCD	Charge-coupled Device
CPU	Central Processing Unit
CR	Computed Radiography
CRT	Cathode Ray Tube
CSV	Colon Separated Values, Semikolon-getrennte Werte in einer Datei
CT	Computed Tomography
DCT	Diskrete Kosinustransformation
DFT	Diskrete Fourier Transformation
DICOM	Digital Imaging and Communications in Medicine
DOS	Disk Operating System (gewöhnlich Microsoft DOS)
DVD	Digital Versatile Disk
GB	Gigabyte
GIF	Graphics Interchange Format
HTTP	Hypertext Transfer Protocol
IEXPLORE	Prozeßname des MS Internet-Explorer, MSIE
IT	Informations Technologie
JPEG	Joint Picture Experts Group
kB	Kilobyte, 1024 Byte
KAS	Klinisches Arbeitsplatzsystem
LAN	Local Area Network
MB	Megabyte
MR	Magnetic Resonance Imaging
MS	Microsoft®, Redmond, WA, U.S.A
MSIE	Microsoft Internet Explorer
NAS	Network Attached Storage
PACS	Picture Archiving and Communication System
PC	Personal Computer
PCI	
RAID	Redundant Array of Independent (oder Inexpensive) Disks
RAM	Random Access Memory
RIS	Radiologie-Informationssystem
S-SCP	Storage Service Class Provider
S-SCU	Storage Service Class Provider
SAN	Storage Area Network
SCP	Service Class Provider
SCSI	Small Computer System Interface
SCU	Service Class User
SP	Service Pack
TB	Terabyte
TCP/IP	Transmission Control Protocol / Internet Protocol
TIFF	Tagged Image File Format
t <sub>rec</sub>	Zeit für den Empfang von Bildern ohne gleichzeitige Umwandlung

$t_{imp}$	Zeit für den Empfang von Bildern mit gleichzeitiger Umwandlung
$t_{pp}$	Zeit für die Nachbearbeitung ohne gleichzeitigen Empfang
TTUp	Zeit für die gesamte Übertragung incl. Nachbearbeitung ( $TTUp = t_{rec} + t_{pp}$ oder $TTUp = t_{imp} + t_{pp}$ )
USB	Universal Serial Bus
VM	Virtual Machine
WAN	Wide Area Network
WORM	Write Once Read Many
WWW	World Wide Web

## 11 Index

### A

- Anwendungsszenarien ..... **13**
- Anzeigevorgang ..... **24, 54**
  - erster nach MSIE-Start ..... **54**
- Anzeigezeit
  - Einflußgrößen ..... **4**
  - gemessene ..... **60**
  - klinisch akzeptable ..... **76**
- API *Siehe* Application Programming Interface
- Applet ..... *Siehe* Java Applet
- Application Programming Interface ..... **34**
- Archivierung
  - Anbindung an das RIS ..... **6**
  - Kosten ..... **9**
  - langfristige ..... **9**
  - traditionelle, filmgebundene ..... **8**
- Automouse ..... **25, 28**

### B

- Befundverteilung ..... **13**
- Bildanzeigezeit .... *Siehe* Anzeigezeit
- Bilder
  - verschiedene vs. gleiche... **42, 53, 84**
- Bildverteilungssoftware ..... **22**
- Bildverteilungssystem ..... **13**
  - Klient-Applet ..... **23**
  - Server-Software ..... **22**
- Browser ..... **10**
- BVS .... *Siehe* Bildverteilungssystem
- Bytecode ..... **19**

### C

- Closed Box Testing ..... **28**
- CR-Bilder
  - heruntergeladene ..... **26**
  - hochgeladene ..... **27**
- CT-Bilder
  - heruntergeladene ..... **26**
  - hochgeladene ..... **27**
- Curator ..... **23**

### D

- Datenvolumen
  - in der Literatur ..... **90**
  - Zunahme ..... **9, 92, 97**

- DCT ..... *Siehe* diskrete Kosinustransformation
- DICOM.... *Siehe* Digital Imaging and Communications in Medicine
- Digital Imaging and Communications in Medicine..... **6**
- digitale Gesundheitskarte ..... **14**
- Diskrete Kosinustransformation ... **15**
  - inverse ..... **15**

### E

- eFilm ..... **21, 25**
- Erstes Bild nach Start ..... **54, 84**
- Exhibit ..... **22**
- Exhibit Java-Applet ..... **23**

### F

- FastEtherChannel ..... **22**
- Fortschrittsicon ..... **24**

### H

- Hochladekapazität ..... **46, 65**
  - erzielte ..... **90**
  - in der Literatur ..... **90**
  - Meßergebnisse ..... **65**
  - Möglichkeiten zur Steigerung .. **92, 97**
- Hochladen
  - Phasen ..... **56, 84**
- Hochladezeit
  - Definition ..... **57, 85**
  - Grafische Darstellung
    - 6 Studien ..... **58**
    - Einzelstudie ..... **57**
    - Vergleich ..... **59**
- HTTP ..... *Siehe* Hypertext Transfer Protocol
- Hypertext Transfer Protocol ..... **10**

### I

- Image Cache ..... **21, 23**
  - während Hochladen ..... **56**
  - 6 Studien ..... **58**
  - Einzelstudie ..... **57**

### J

- Java ..... **19**
  - Applet ..... **12, 19**
- Joint Picture Experts Group ..... **15**

JPEG ..... <i>Siehe</i> Joint Picture Experts Group	Qualität.....	<b>49</b>
<b>K</b>	Schaltplan.....	<b>31</b>
KAS .... <i>Siehe</i> Klinisches Arbeitsplatz System	Methode 3	
Kleiner Server	Ablauf-Schema.....	<b>37</b>
Hardwaredefinition.....	Beschreibung .....	<b>34</b>
<b>45</b>	Bewertung	
Klient	Gebrauch .....	<b>82</b>
Hardware	Qualität .....	<b>79</b>
ausreichende.....	Gebrauchsfähigkeit .....	<b>51</b>
Modifikationen .....	Qualität.....	<b>49</b>
Software.....	Scriptsprache .....	<b>39</b>
<b>23</b>	Microsoft Internet Explorer.....	<b>11, 23</b>
Klient-PC .....	Microsoft Virtual Machine.....	<b>23</b>
21, <b>25</b>	MR-Bilder	
KliNet.....	heruntergeladene .....	<b>26</b>
52 <i>Siehe</i> Netzwerk, verwendetes	hochgeladene.....	<b>27</b>
Klinisches Arbeitsplatz System.....	MSIE ..... <i>Siehe</i> Microsoft Internet Explorer	
Kompressionsverfahren .....	<b>15</b>	
<b>L</b>	<b>N</b>	
LAN .....	Nachbearbeitung. 46, <b>57, 58, 91, 97, 116</b>	
<i>Siehe</i> Local Area Network	Definition .....	<b>23</b>
Leistungsindikator .....	durch Serverfarm.....	<b>98</b>
<b>28</b>	Grafische Darstellung	
beobachtete	6 Studien.....	<b>58</b>
Methode 1 .....	Einzelstudie .....	<b>57</b>
<b>30</b>	Vergleich.....	<b>59</b>
Server.....	Netzwerk	
<b>42</b>	geswitcht .....	<b>5</b>
Protokollsoftware .....	Klient-Hardware-Tests.....	<b>44</b>
<b>29</b>	nicht-geswitcht.....	<b>5</b>
Link.....	verwendetes .....	42, 52, 83, 88
<b>11</b>	<b>P</b>	
Local Area Network .....	PACS ... <i>Siehe</i> Picture Archiving and Communication System	
<b>22</b>	perfmon.....	<b>28</b>
<b>M</b>	Phototransistoren.....	<b>31</b>
Material und Methoden.....	Picture Archiving and Communication System .....	<b>8</b>
<b>21</b>	Kosten .....	<b>9</b>
MB .....	Pixel	
<i>Siehe</i> Megabyte	überwachte.....	<b>37</b>
<b>22</b>	CR-Einzelbild .....	<b>38</b>
Methode 1	CT/MR-Tableau .....	<b>39</b>
Ablauf-Schema .....	Plugins .....	<b>12</b>
<b>29</b>	Primärkompression .....	<b>23, 24</b>
Beschreibung.....	verlustbehaftete.....	<b>23</b>
<b>28</b>	verlustfreie.....	<b>23</b>
Bewertung		
Gebrauch .....		
<b>80</b>		
Qualität.....		
<b>79</b>		
Gebrauchsfähigkeit.....		
<b>51</b>		
Qualität .....		
<b>49</b>		
Methode 1b		
Beschreibung.....		
<b>30</b>		
Bewertung.....		
<b>79</b>		
Qualität .....		
<b>49</b>		
Methode 2		
Ablauf-Schema .....		
<b>33</b>		
Beschreibung.....		
<b>31</b>		
Bewertung		
Gebrauch .....		
<b>81</b>		
Qualität.....		
<b>79</b>		
Gebrauchsfähigkeit.....		
<b>51</b>		

<b>Q</b>	
Quantisierungstabelle.....	16
<b>R</b>	
Radiologie-Informationssystem .....	6
RAID..... <i>Siehe</i> Redundant Array of Independent Disks	
RAM .....	<i>Siehe</i> Random Access Memory
Random Access Memory	
Klient-Hardware-Tests .....	44
Klient-PC.....	25
Sender-PC .....	25
Server .....	22
Redundant Array of Independent Disks	
im Server .....	22
Referenzbilddatenbank .....	14
RIS .....	<i>Siehe</i> Radiologie-Informationssystem
<b>S</b>	
SCP ... <i>Siehe</i> Service Class Provider	
Script	
Automouse.....	28
Methode 1.....	29
Methode 2.....	31
Methode 3.....	35
Scriptsprache für Methode 3....	39
Sender-PC .....	25
SCU..... <i>Siehe</i> Service Class User	
Sekundärkompression.....	24
verlustbehaftete .....	24
Empfehlung .....	88, 96
Sender-PC.....	21, 25
Server	
Hardware .....	22
kleiner .....	45
Modifikationen.....	45
Software.....	22
Standard .....	22
Überwachungsanwendung .....	42
Service Class Provider .....	7
Service Class User .....	7
S-SCP <i>Siehe</i> Storage Service Class Provider	
Standardserver .....	22
Storage Service Class Provider ...	23
Studienaufbau.....	21
<b>T</b>	
Tableau.....	23, 27
Teleradiologie .....	14
Testaufbau .....	21
Testkoordinator-PC.....	25, 34
Testserien .....	43
$t_{imp}$ .....	85
Definition .....	58
$t_{pp}$ .....	<i>Siehe</i> Nachbearbeitung
$t_{rec}$ .....	85
Definition .....	57
TTUp..... <i>Siehe</i> Hochladezeit	
<b>V</b>	
verschiedene Bilder .....	<i>Siehe</i> Bilder, verschiedene vs. gleiche
Viewing Station .....	8, 10
Virtual Machine .....	19
Vortests	
Beschreibung .....	42
Ergebnisse .....	52
Vorverarbeitung von Bildern.....	12
<b>W</b>	
WAN .....	<i>Siehe</i> Wide Area Network
Wavelet.....	17
Web Cache	
während Hochladen .....	57
Web Cache .....	21, 23
Webbasierte Verfahren .....	10, 77
Webtechnologie .....	10
Wide Area Network	
Anwendungsszenario .....	14
Empfehlung .....	96
Empfehlungen für .....	89
Workflow .....	9, 10, 13
World Wide Web .....	10
<b>Z</b>	
Ziel .....	4
Zugriffszahlen	
realistische .....	77

## 12 Danksagung

Danken möchte ich Herrn Prof. Dr. Th. Vogl und Prof. Dr. Björn Bergh, die mir diese Arbeit ermöglichten.

Des Weiteren dem Team des Dezernats für Informations- und Kommunikationstechnologie der Uniklinik Frankfurt, allen voran Herrn Andreas Schlaefke, der mir in vielen Situationen hilfreich zur Seite stand, fleißig PCs für die Tests zusammenschraubte und zusammen mit mir viele Tage mit der Durchführung der Testserien verbrachte. Mein Dank gilt auch Conny Stahl, die immer wieder lange Bilderlisten für die Versuchsreihen bereitstellte.

Meinen Freunden, vor allem den RCDS Bundesvorständen Barbara von Wnuk-Lipinski, Stefan Glösenkamp, Benjamin Feindt und Dorlies Last sowie den Hessischen Landesvorsitzenden Thilo Schmidt und Marian Zachow, die mich immer wieder motiviert und unterstützt haben und Herrn Norbert Beckmann-Dierkes vom Bildungszentrum Wendgräben der Konrad-Adenauer-Stiftung. Und natürlich Heidi Dobberstein für die Rechtschreibkorrektur.

Besonderer Dank gilt auch dem Team der Station 142 der Charité Campus Mitte, insbesondere Herrn Prof. Lochs, Herrn Dr. Pirlich und Frau Dr. Schelwies, die den schreibenden Patienten immer wieder motiviert haben.

Hier sind auch besonders meine Eltern und Großeltern zu nennen, die bereits in frühen Jahren mein Interesse an wissenschaftlichem Arbeiten weckten und förderten.



## 13 Schriftliche Erklärung und Publikationsverzeichnis

Ich erkläre, daß ich die dem Fachbereich Medizin der Johann Wolfgang Goethe-Universität Frankfurt am Main zur Promotionsprüfung eingereichte Dissertation mit dem Titel

„Webbasierte Bildverteilung

in der Radiologie:

Performanzmessungen und Anforderungen an die Hardware“

in dem Institut für Diagnostische und Interventionelle Radiologie unter Betreuung und Anleitung von Prof. Dr. med. Th. Vogl mit Unterstützung durch Herrn Prof. Dr. Bergh ohne sonstige Hilfe selbst durchgeführt und bei der Abfassung der Arbeit keine anderen als die in der Dissertation angeführten Hilfsmittel benutzt habe.

Ich habe bisher an keiner in- oder ausländischen Universität ein Gesuch um Zulassung zur Promotion eingereicht.

Die vorliegende Arbeit wurde bisher nicht als Dissertation eingereicht.

Die vorliegende Arbeit wurde in Teilen in folgenden Publikationsorganen veröffentlicht:

1. Bergh B, Pietsch M, Vogl TJ (2000) Internettechnologie für klinische Anwendungen in einer digitalen Röntgenabteilung. Radiologe. Apr;40(4):371-87
2. Bergh B, Schlaefke A, Pietsch M, Garcia I, Vogl TJ (2003) Evaluation of a "no-cost" Internet technology-based system for teleradiology and co-operative work. Eur Radiol. Feb;13(2):425-34
3. Bergh B, Pietsch M, Schlaefke A, Garcia I, Vogl TJ (2003) Performance of Web-based image distribution: server-oriented measurements. Eur Radiol. Nov;13(11):2419-24
4. Bergh B, Pietsch M, Schlaefke A, Vogl TJ (2003) Performance of Web-based image distribution: client-oriented measurements. Eur Radiol. Sep;13(9):2161-9
5. Bergh B, Pietsch M, Schlaefke A, Garcia I, Vogl TJ (2004) Upload capacity and time-to-display of an image Web system during simultaneous up- and download processes. Eur Radiol. Mar;14(3):526-33.
6. Pietsch M, Schlaefke A, Vogl TJ, Bergh B Development and evaluation of different methods to assess download- and display-time of image web systems. Eingereicht bei Journal of Digital Imaging

Frankfurt am Main, den 07.10.2005

## 14 Tabellarischer Lebenslauf

Persönliche Daten Michael Pietsch  
Adalbert-Stifter-Straße 9  
69221 Dossenheim  
E-Mail: m.pietsch@gmx.de

Geburtsdatum 09. April 1976  
Geburtsort Frankfurt am Main

Familienstand ledig  
Staatsangehörigkeit deutsch  
Eltern Dr. Eva Pietsch-Berger, Ärztin für Allgemeinmedizin  
Dr. Ullrich Pietsch, Arzt für Innere Medizin

### Schulbildung

1982 - 1986 Grundschule, Glauburg-Stockheim  
1986 - 1992 Gesamtschule Konradsdorf, Ortenberg  
01/1990 - 09/1990 USA-Aufenthalt, Gettysburg, PA,  
Gettysburg Junior High School  
1992 - 1995 Oberstufe am Wolfgang-Ernst-Gymnasium zu Büdingen  
Schulabschluß Abitur, Durchschnittsnote 2,0

### Hochschulbildung

10/1995 - 05/2002 Studium der Humanmedizin an der Johann  
Wolfgang Goethe-Universität, Frankfurt am Main

10/1997 Physikum, Durchschnittsnote 4,0

10/1999 1. Staatsexamen, Durchschnittsnote 4,0

05/2001 2. Staatsexamen, Durchschnittsnote 3,3

05/2001 - 05/2002 Praktisches Jahr

05/2002 3. Staatsexamen, Note 1,0

seit 10/2004 Aufbaustudiengang „Informationsmanagement in der  
Medizin“  
Ruprechts-Karls-Universität Heidelberg  
Fachhochschule Heilbronn

### Beruf

06/2002 - 10/2002 Wissenschaftlicher Mitarbeiter BAT IIa/2 im Dezernat  
für Informations- und Kommunikationstechnologie der  
Uniklinik Frankfurt am Main

11/2002 - 11/2004 Bundesgeschäftsführer des Rings Christlich-Demokra-  
tischer Studenten

Frankfurt am Main, den 07.10.2005



