# Konzeption und Realisierung eines Verteilten Master Patient Index

Diplomarbeit im Fach Informatik

vorgelegt von

**Gernot Roth**

geb. 02.08.1981 in Nürnberg

angefertigt am

**Department Informatik**
**Lehrstuhl für Informatik 6**
**Datenmanagement**
**Friedrich-Alexander-Universität Erlangen-Nürnberg**

Betreuer: Prof. Dr. Richard Lenz
Christoph P. Neumann

Beginn der Arbeit: 02.01.2009
Abgabe der Arbeit: 30.06.2009

# Erklärung zur Selbständigkeit

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass diese Arbeit in gleicher oder Ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Der Friedrich-Alexander-Universität Erlangen-Nürnberg, vertreten durch den Lehrstuhl für Informatik 6 (Datenmanagement), wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Diplomarbeit einschließlich etwaiger Schutzrechte und Urheberrechte eingeräumt.

Erlangen, den 30.06.2009

(Gernot Roth)

# Kurzzusammenfassung

In Untersuchungen zu medizinischen Behandlungsfehlern treten immer wieder Probleme bei der Re-Identifizierung von Patienten auf. Daher bemühen sich Informatiker daran beteiligte Prozesse durch IT zu unterstützen. Es existieren bereits Konzepte, die durch Nutzung von zentralen Diensten eine Zuordnung von Patienteninformationen unterschiedlicher Organisationseinheiten ermöglichen. Um mit diesen Konzepten eine bessere Skalierung zu realisieren, soll ein verteiltes System entworfen werden, mittels dessen die Funktionen dieser Konzepte auch in Abwesenheit eines zentralen Dienstes umgesetzt werden können. Hierfür werden Methoden zur Anonymisierung von Patientendaten genutzt, auf deren Ergebnissen noch eine verlässliche Zuordnung von Datensätzen identischer Patienten möglich ist. Des Weiteren wurde ein Mechanismus eingesetzt, mit dessen Hilfe eine effiziente Suche in dem Datenbestand des Gesamtsystems ermöglicht wird. Das Konzept zur Realisierung eines solchen verteilten 'Master Patient Index' wird in dieser Arbeit dargestellt.

# Abstract

Investigations about types of medical errors consistently show problems in the patients identification process. Therefore computer scientists try to support the participating processes by IT. Concepts exist already which allow correlation of patient related information using centralized services. For enhancing the scaling ability of these concepts, a distributed system has to be designed, which is capable to fulfil the tasks of these concepts even in absence of a central node. For this purpose anonymisation-methods are used, which still allow the correlation of the data. Furthermore it uses a design, which allows efficient search operation over the complete system. The concept of such a 'distributed master patient index' is presented in this thesis.

# Inhaltsverzeichnis

**1. Einleitung** **1**

**2. Methodik** **3**

**3. Grundlagen** **5**

**4. Stand der Technik** **7**
4.1. PIX . . . 7
4.1.1. Anforderungen . . . 8
4.1.2. Funktionsweise . . . 9
4.1.3. Transaktionen . . . 11
4.2. PIDS . . . 15
4.2.1. Aufbau . . . 16
4.2.2. Interfaces . . . 17
4.2.3. ConformanceClasses . . . 21
4.3. MPI Implementierungen . . . 22
4.3.1. promedtheus MPI . . . 22
4.3.2. Sun MPI . . . 24
4.4. Zusammenfassung . . . 26

**5. Verwandte Arbeiten** **29**
5.1. Phonetische Algorithmen . . . 29
5.1.1. Soundex . . . 29
5.1.2. Metaphone . . . 30
5.1.3. NYSIIS . . . 31
5.1.4. Kölner Phonetik . . . 31
5.1.5. Alternativen . . . 31

**6. Lösungsansatz** **33**
6.1. Standardisierung . . . 36
6.2. Phonetische Kodierung . . . 39
6.3. Hashing . . . 39
6.3.1. Hashing über einzelne Attribute . . . 40
6.3.2. Hashing über Attributsgruppen . . . 41

6.4. Kommunikationsverfahren . . . . . 43
6.4.1. Inhomogene Netzstruktur . . . . . 46
6.4.2. Use-Case . . . . . 50
6.5. Zusammenfassung . . . . . 50

**7. Zusammenfassung** **53**

**Abbildungsverzeichnis** **59**

**Tabellenverzeichnis** **61**

**A. Anhang** **63**

**Literaturverzeichnis** **65**

# 1. Einleitung

In der heutigen Medizin gibt es verschiedene Untersuchungen zur Ursache von Behandlungsfehlern. Dabei spielen Fehler, die durch die falsche Identifizierung eines Patienten entstehen immer eine wichtige Rolle. Um diese zu vermeiden, befassen sich verschiedene Projekte in der Informatik mit der Verbesserung der in die Identifizierung involvierten Vorgänge. Zum einen soll verhindert werden, dass Behandlungen die für einen Patienten geplant sind, an eigentlich unbeteiligten Personen durchgeführt werden. Ist dies der Fall sind bereits im Vorfeld eine Kette von Fehlern aufgetreten [CB02]. Ein weiterer Grund für fehlerhafte Behandlungen, deren Ursachen mit der Identifizierung zusammen hängen, kann auch die nicht erfolgreiche Re-Identifizierung eines Patienten sein, wodurch Vorbefunde und bekannte Unverträglichkeiten nicht berücksichtigt werden können. Im Rahmen dieser Problematik gibt es Forschungsprojekte innerhalb der Informatik, auf welche Arten die betroffenen Prozesse in der Medizin durch eine IT Infrastruktur unterstützt werden können um die Fehlerquote zu reduzieren. Zusätzlich kann die Qualität einer medizinischen Behandlung weiter verbessert werden, je mehr Informationen dem Behandelnden zur Verfügung stehen. Somit ist auch die Aufbereitung der Daten und deren Bereitstellung am Behandlungsort eine in diesem Zusammenhang relevante Aufgabe der IT. Es werden also Systeme gebraucht, die es einem Arzt zum Beispiel ermöglichen vom Behandlungsraum aus auf Laborergebnisse zuzugreifen.

Bei bestehenden Krankenhausinformationssystemen (KIS) herrscht oft eine gewachsene Infrastruktur, wodurch meist keine einheitliche Formatierung der identifizierenden elektronischen Daten einen Patienten betreffend vorhanden ist. Die Identifikationsnummern für einen Patienten können in den verschiedenen administrativen Einheiten variieren, so dass es nötig ist, Systeme zu entwickeln, die die Vermittlungsarbeit zwischen den einzelnen Domänen leisten. Zu diesem Zweck wurden diverse Master Patient Index (MPI) Systeme entworfen. Diese dienen der Verknüpfung der verschiedenen Einheiten innerhalb des Krankenhauses, es müssen allerdings alle dazu nötigen (nichtmedizinischen) Daten zentral in einem System (dem MPI) zur Verfügung gestellt werden. Dadurch ist es nicht möglich, diese Lösung auf beliebig große Mengen von teilnehmenden Systemen auszuweiten. Zum einen entstehen hierbei rechtliche Probleme im Zuge der Datensicherheit und des Datenschutzes. Weiterhin kann die freie Skalierbarkeit auf Grund des zentralen Speichers nicht realisiert werden. Daher soll eine Lösung gefunden werden, die die Möglichkeit eröffnet, eine elektronische Patientenakte automatisch allen an der Behandlung des Patienten be-

teiligten Einheiten zur Verfügung zu stellen, wobei sowohl die Echtheit als auch die Korrektheit der Daten sichergestellt sein muss.

Es wird ein Konzept zur Realisierung eines verteilten Master Patient Index erstellt. Dessen Ziel ist es, den teilnehmenden Systemen die Möglichkeit zu geben, auch ohne Kenntnis der Systeme die Daten über den betroffenen Patienten besitzen, an die ID's der für diese Anfrage relevanten Systeme, inklusive der dortigen ID des Patienten zu gelangen. Anschließend können mit Hilfe dieser Daten die medizinischen Informationen zu diesem Patienten abgefragt werden, um dem Behandelnden eine umfassende elektronische Patientenakte zur Verfügung zu stellen. Hierzu werden im Rahmen dieser Arbeit einige bestehende MPI Systeme und vorhandene Konzepte zur Patientenidentifizierung beleuchtet. Diese basieren jedoch auf gewissen hierarchischen Strukturen, und stellen zum Teil strenge Anforderungen an die teilnehmenden Systeme, so dass die Ziele eines verteilten MPI nicht allein durch die bestehenden Konzepte realisiert werden können. Es werden Lösungsansätze diskutiert und bewertet, mit Hilfe derer die Autonomie der teilnehmenden Systeme gewahrt bleiben soll, aber gleichzeitig eine umfassende Informationsbeschaffung zu einem Patienten ermöglicht wird, indem die dazu nötigen Patienten ID's zur Verfügung gestellt werden.

# 2. Methodik

Um die Anforderungen an einen möglichen Ansatz zur Realisierung eines verteilten Master Patient Index genauer spezifizieren zu können müssen zunächst die Anforderungen an normale MPI Systeme untersucht werden. Zusätzlich zu diesen muss definiert werden welche weiteren Problemstellungen hinzu kommen wenn der zentralisierte Ansatz auf ein verteiltes System übertragen werden soll. Zu diesem Zweck werden einige Spezifikationen bestehender MPI Systeme genau untersucht und die verwendeten Werkzeuge werden im Hinblick auf einen möglichen Einsatz in einem verteilten MPI hin bewertet. Anschließend werden einige Methoden vorgestellt die genutzt werden um die zusätzlichen Anforderungen an einen verteilten MPI erfüllen zu können.

Die verschiedenen Kombinationsmöglichkeiten der Parameter die zur Lösung der Probleme führen können werden bewertet und die effizientesten Varianten werden dargestellt. Weiterhin wird ein Entwurf gezeigt mit dessen Hilfe die Kommunikation zwischen den einzelnen teilnehmenden Systemen geregelt werden kann, welcher verbesserte Ergebnisse hinsichtlich der Performance des Gesamtsystems liefert. Abschließend wird das Ergebnis der Arbeit kurz zusammengefasst und ein Ausblick auf Einsatzmöglichkeiten und weitere Entwicklungen gegeben.

# 3. Grundlagen

In diesem Bereich gehe ich darauf ein, wie ein zentralisiertes MPI System arbeitet, und warum dessen Funktionalität in einem verteilten Umfeld nicht ohne weiteres zu erhalten ist. Einem MPI werden von allen teilnehmenden Subsystemen Informationen zur Verfügung gestellt, anhand derer es möglich ist, die betroffene Person eindeutig zu bestimmen. Werden nun die Datensätze von Vergleichsalgorithmen verarbeitet, und dabei hinreichende Übereinstimmungen innerhalb von verschiedenen Datensätzen, die darauf hindeuten das es sich um denselben Patienten handelt, gefunden, so wird eine Verknüpfung dieser Daten erstellt. Sollte nun ein System nach Informationen zu einem Patienten suchen, kann dieses eine Anfrage an den MPI stellen, und dieser versorgt das System mit den benötigten Identifikationsnummern die der Patient in den anderen administrativen Einheiten hat. Hierdurch wird eine direkte Anfrage an ein anderes System mit der dortigen ID des Patienten ermöglicht.

Ein MPI unterscheidet sich hierbei von einer elektronischen Patientenakte dadurch, dass lediglich die verschiedenen Schlüssel und identifizierenden Merkmale gespeichert werden, nicht jedoch die gesamten medizinischen Daten, die den Patienten betreffen. Bei der Erzeugung von Verknüpfungen zwischen verschiedenen ID's muss sichergestellt werden, dass keine medizinisch kritischen Daten einem falschen Patienten zugeordnet werden können. Daher ist es vorteilhaft für diese Operation eine möglichst große Menge an Identifikationsmerkmalen aus jedem Teilsystem zur Verfügung zu haben. Wenn ein MPI System aber nicht mehr intern innerhalb eines KIS betrieben wird, treten sicherheitskritische Probleme auf. Die Menge an identifizierenden Daten, die einem System für die Vergleichsoperation zur Verfügung steht, muss eingeschränkt werden. Ebenso wenig dürfen diese Daten an nicht autorisierte Stellen gelangen, oder Daten von nicht vertrauenswürdigen Quellen verarbeitet werden. Es gibt hierbei verschiedene Varianten, wie der Verknüpfungsprozess ablaufen kann. Zum einen existieren vollautomatisierte Systeme, die bei Erfüllung gewisser Anforderungen an den Grad der Gleichheit automatisch die Verknüpfung der Datensätze im Hintergrund ausführen, in anderen Systemen werden lediglich potentielle Treffer gemeldet und die eigentliche Verknüpfung geschieht manuell durch einen autorisierten Benutzer.

Ein weiteres Problem bei der Konzeption einer solchen Lösung ist die stark variierende Beschaffenheit der bestehenden Systeme. Es gibt eine große Anzahl unterschiedlicher Datenbanken in denen die aktuell vorhandenen Daten gespeichert sind. Ebenso unterscheiden sich die Art der vorhandenen Daten bei Fachärzten verschiede-

ner Spezialbereiche, wobei auch vorhandene Standards wie der HL7 noch nicht von allen, zum Teil älteren Systemen eingehalten werden. Des Weiteren hat die Praxissoftware eines niedergelassenen Arztes einen vollkommen anderen Schwerpunkt als ein großes KIS System, und auch die zahlreich vorhandenen Schnittstellen, die einer Applikation an einem KIS zur Verfügung stehen, darf man in diesem Kontext nicht als Voraussetzung betrachten, da hierdurch viele kleinere Praxen von der Teilnahme an der gemeinsamen Nutzung von Informationen ausgeschlossen werden würden. Man kann einen MPI als ein abgeschlossenes autonomes System ansehen. In diesem Fall kann man das Ziel eines verteilten MPI als eine Spezialisierung eines Verteilten Systems betrachten. Als Einstufung eines solchen schreiben Tanenbaum/Steen[AST08]:

> "Ein verteiltes System ist eine Ansammlung unabhängiger Computer, die den Benutzern wie ein einzelnes kohärentes System erscheinen."

Die in dieser Aussage angesprochenen unabhängigen Computer sind in diesem Kontext die einzelnen Patientenindizes in den Subsystemen, die sich bei Anfragen wie ein einziger Datenspeicher verhalten und alle Ergebnisse zu der Anfrage zurück liefern sollen. Dies geschieht dabei unabhängig von der administrativen Einheit, in der die Daten gespeichert waren. Dieses Verhalten soll, ohne den Einsatz eines zentralen Servers in dem alle ID's gespeichert werden, umgesetzt werden, um die zuvor angesprochenen Probleme bezüglich Sicherheit und Skalierung zu vermeiden. Ein umfangreiches Rechtekonzept ist allerdings auch in einer solchen Umgebung notwendig, in dem beispielsweise die Autorisierung auf den Systemen und die Authentifizierung der Systeme zueinander geregelt ist.

# 4. Stand der Technik

In diesem Kapitel möchte ich bestehende Ansätze im Bereich der Patientenidentifizierung zusammenfassen. Der Schwerpunkt liegt dabei auf zwei Projekten, dem 'Patient Identifier Cross-referencing Integration Profile' (PIX) von der IHE (Integrating the Healthcare Enterprise) [ACC07a], und dem 'Person Identification Service' (PIDS) [Obj01] von der 'Object Management Group' (OMG). Hierbei stellt PIX ein Integrationsprofil dar, welches es ermöglichen soll mit Hilfe von Querverweisen über unterschiedliche administrative Bereiche, die eigene Identifizierungsmittel benutzen, hinweg auf Patientendaten zuzugreifen. Hierfür existiert eine zentrale Stelle die diese Querverweise verwaltet und von der aus die domänenübergreifenden Zugriffe reglementiert und realisiert werden können. Beim zweiten betrachteten Projekt PIDS werden Schnittstellen definiert, über die es ermöglicht werden soll plattform- und applikationsunabhängig Kommunikation zwischen bestehenden Systemen zu ermöglichen. Dieses System stellt außerdem Dienste zur Verfügung, mit Hilfe derer sowohl ID's vergeben werden können, als auch die Bildung von Querverweisen zwischen Datensätzen aus unterschiedlichen Domänen möglich ist, sofern diese Datensätze derselben Person entsprechen.

## 4.1. PIX

Ziel des 'Patient Identifier Cross-referencing Integration Profile' von IHE ist es, die Möglichkeit zu schaffen aus einer IHE Patient Identifier Domain an Informationen zu einem bestimmten Patienten zu gelangen die in einer anderen IHE Patient Identifier Domain vorhanden sind. Da jede dieser Domänen potentiell eigene Primärschlüssel als Patienten-ID's vergibt, ist es zunächst nötig die dortige ID des Patienten über den Informationen abgefragt werden sollen zu ermitteln. Eine solche IHE Patient Identifier Domain stellt also eine abgeschlossene, administrative Einheit mit einer beliebigen Anzahl von Systemen die miteinander kommunizieren können dar. Diese Systeme nutzen alle ein gemeinsames Identifikationsschema, wobei es genau ein festgelegtes System gibt, das sowohl dafür verantwortlich ist eine einzigartige ID für jedes, einen Patienten betreffende Objekt, zu generieren, sowie eine Auswahl von Identitätseigenschaften des Patienten zu pflegen. Dieses System wird IHE Patient Identity Source genannt und ist in jeder IHE Patient Identifier Domain einzigartig. Des Weiteren enthält jede Domäne mindestens ein System, welches die Querverweise auf ID's ei-

nes Patienten in anderen Domänen anfordert beziehungsweise entgegennimmt. Dieses wird als IHE Patient Identifier Cross-reference Consumer bezeichnet. Alle weiteren Systeme die noch Bestandteil einer IHE Patient Identifier Domain sein können stehen mit diesen Systemen über beliebige domäneninterne Schnittstellen in Verbindung, auf die im Rahmen dieses Integrationsprofils nicht näher eingegangen wird, da hierbei lediglich die Kommunikation über Domänengrenzen hinweg betrachtet wird. Diese läuft immer über ein zentrales System, welches mit den einzelnen IHE Patient Identity Source und IHE Patient Identifier Cross-reference Consumer kommunizieren kann. Dieses zentrale System des Profils, der IHE Patient Identifier Cross-reference Manager (CRM), ist genau einmal vorhanden, und bildet zusammen mit N IHE Patient Identifier Domains die übergeordnete IHE Patient Identifier Cross-reference Domain, was in Abbildung 4.1[ACC07a] veranschaulicht wird.

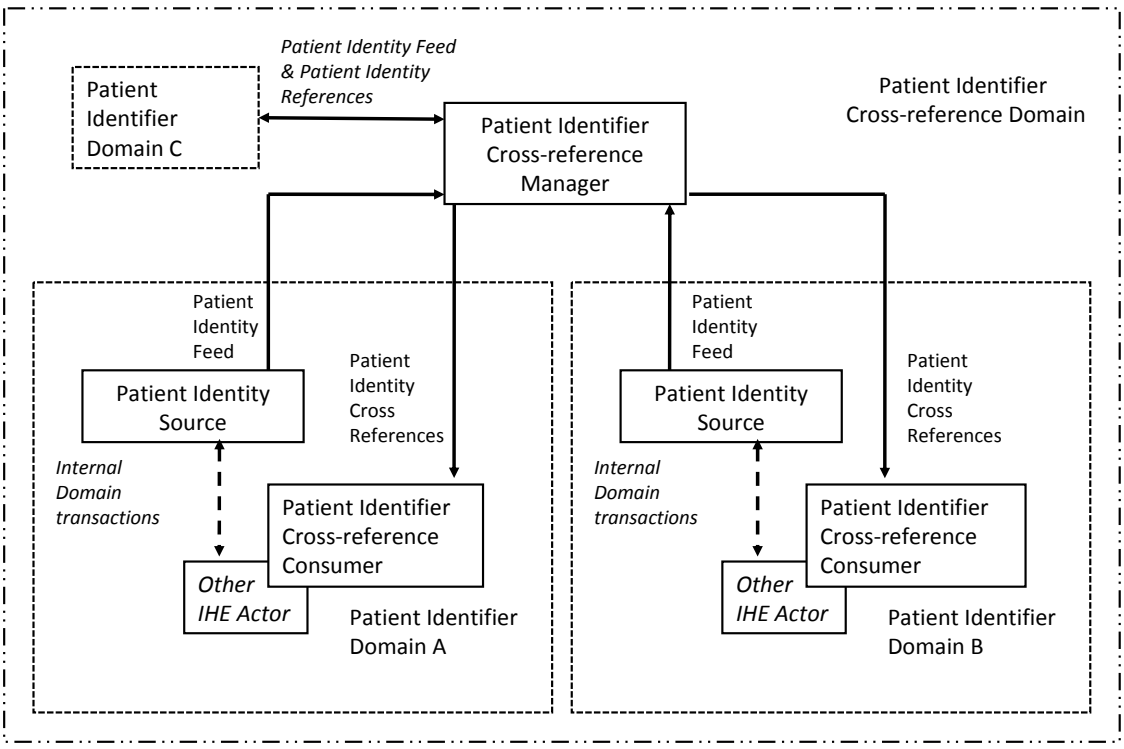


Abbildung 4.1.: Process Flow with Patient Identifier Cross-referencing

### 4.1.1. Anforderungen

Zur Realisierung dieses Integrationsprofils wird an jede der teilnehmenden Domänen eine Reihe von Anforderungen gestellt. So gibt es Bedingungen an die interne Verwaltung einer teilnehmenden Domäne. Wie bereits beschrieben muss es genau ein

System in jeder Domäne geben das neue ID's generieren darf. Es muss ein Regelsatz existieren, welcher definiert, wie innerhalb dieser Domäne ID's festgelegt sind, und wie diese verwaltet werden um den speziellen Anforderungen dieser Domäne (z.B. Radiologie) gerecht zu werden. Innerhalb der Domäne muss eine Administrationsstelle existieren um eben diesen Regelsatz zu regulieren. Weiterhin muss jede Domäne einen, innerhalb der IHE Patient Identifier Cross-reference Domain einzigartigen, Bezeichner haben um Daten anhand der Kombination aus Domänenbezeichner und Objekt-ID eindeutig zuordnen zu können. Auch wenn idealer weise jeder Patient in einer Domäne nur eine ID zugewiesen bekommt, unterstützt PIX, sofern dies dem CRM bekannt gemacht wird, zusätzlich die Möglichkeit einem Patienten multiple ID's innerhalb einer Domäne zuzuweisen.

Im Umfang der IHE Patient Identifier Cross-reference Domain gibt es noch weitere Anforderungen an die einzelnen Domänen untereinander. Es wird davon ausgegangen, dass sich die Domänen auf einen Regelsatz geeinigt haben, welcher festlegt, wie Patienten-ID's über einzelne Domänengrenzen hinweg referenziert werden können. Die Domänen müssen sich auf Verwaltungsprozesse einigen, die genutzt werden um diesen Regelsatz zu administrieren. Es muss eine Instanz geben die das Recht hat diesen Regelsatz und die Prozesse zu verwalten. Im Rahmen von PIX wird ein möglichst großer Teil der funktionalen Bedingungen in den CRM zentralisiert um die Anforderungen an die einzelnen teilnehmenden Domänen so gering wie möglich zu halten. Sind diese Mindestanforderungen nicht erfüllt kann eine Implementierung nicht die volle Funktionalität des Integrationsprofils ausschöpfen.

### 4.1.2. Funktionsweise

Definierte Transaktionen zwischen den einzelnen Aktoren, sind zum einen die Übertragung von Identitätsinformationen über Patienten von einer IHE Patient Identity Source zum IHE Patient Identifier Cross-reference Manager, ein sogenanntes IHE Patient Identity Feed. Zum Anderen muss der IHE Patient Identifier Cross-reference Consumer die Möglichkeit haben, eine Liste mit Referenz-ID's aus anderen Identifikationsdomänen zu bekommen. Hierfür werden in dem Profil zwei Möglichkeiten dargestellt, entweder stellt der IHE Cross-reference Consumer eine Anfrage an den CRM und erhält darauf eine Antwort, oder dieser Vorgang wird über eine IHE Update Notification, also push-basiert, realisiert.

Das Integrationsprofil weist lediglich diese Transaktionen auf die jeweiligen Aktoren zu, definiert aber nicht über welchen Algorithmus die Referenzen erzeugt werden und legt auch keine speziellen Regeln bezüglich der Inhalte fest die hier zwischen den einzelnen Domänen ausgetauscht werden. Dieses Verhalten wird komplett im CRM realisiert, wodurch sowohl die Kompatibilität zu verschiedenen zugrundeliegenden Informationssystemen gewährleistet werden soll, als auch die Möglichkeit geboten wird die Regeln und den Algorithmus zur Steuerung der domänenübergreifenden

Kommunikation flexibel zu wählen.

Als IHE Patient Identifier Domain wird ein Bereich bezeichnet, in dem alle Systeme das gleiche Identifikationsschema für Patienten verwenden und diese sich auch an die gleiche Stelle, die IHE Patient Identity Source, wenden um Patienten-ID's generieren zu lassen. Im Integrationsprofil einer solchen Domäne außerdem werden einige Eigenschaften festgelegt. Zum einen gibt es einen Regelsatz, in dem beschrieben wird, wie innerhalb dieser Domäne ID's generiert und verwaltet werden. Dieser Regelsatz kann an die speziellen Anforderungen dieser Domäne, und damit einem spezifischen Teilbereich des Gesamtsystems, angepasst werden. Für jede Domäne muss außerdem eine Stelle definiert sein, an der die internen Regeln domänenweit administriert werden können. Des Weiteren muss es eine einzige genau festgelegte Stelle innerhalb der Domäne geben, welche das alleinige Recht besitzt einzigartige ID's zu erzeugen und jedes neue Objekt einen Patienten betreffend mit einer solchen zu versehen. Auch wird an dieser Stelle zu jedem Patienten eine Reihe von Identifikationsmerkmalen gespeichert. Alle anderen Systeme innerhalb der Domäne verlassen sich bezüglich der ID's auf die vergebende Stelle. Im Idealfall wird hier jedem Patienten innerhalb einer Domäne nur einmal eine einzigartige ID zugewiesen. Es ist jedoch auch kein Problem mehrere ID's für den gleichen Patienten bereitzustellen dies muss dann allerdings beim propagieren der Information an den CRM mit angegeben werden.

Jede dieser Domänen besitzt eine Domänen-ID die innerhalb einer IHE Patient Identifier Cross-referencing Domain einzigartig sein muss. Eine solche IHE Cross-referencing Domain besteht aus einer Menge von IHE Patient Identifier Domains und wird von einem CRM verwaltet. Dieser CRM führt Listen in denen die Informationen zu einzelnen Patienten aus verschiedenen Domänen miteinander verknüpft sind. Man könnte sagen es werden hier Übersetzungstabellen der ID's aus den verschiedenen IHE Patient Identifier Domains geführt.

Die IHE Patient Identifier Cross-reference Domain beinhaltet, dass die teilnehmenden Identifier Domains sich über einen Regelsatz geeinigt haben, wie Patienten ID's über die einzelnen Domänengrenzen hinweg referenziert werden können. Weiterhin gibt es eine festgelegte Auswahl an Prozessen mit deren Hilfe dieser Regelsatz erweitert bzw. editiert werden kann und es existiert eine Instanz welche die Verwaltung dieser Regeln und Prozesse übernimmt. Zwar werden möglichst geringe Anforderungen an die teilnehmenden Domänen gestellt, indem man einen Großteil der Funktionsbedingungen im CRM zentralisiert, jedoch müssen zumindest diese Einigungen erzielt worden sein wenn man das Profil erfolgreich implementieren will. Der CRM ist hierbei nicht für die Datenqualität verantwortlich. Es wird vielmehr davon ausgegangen, dass die Identitätsdaten, die von den Quellen an diesen übermittelt werden, bereits von hoher Qualität sind. Die Idee hierbei ist, die Verantwortlichkeit über die Qualität und Verwaltung ihrer demografischen Informationen, sowie die Integrität der ID's die in diesen verwendet werden, in den einzelnen Domänen, und damit ihrer IHE Source Actor, zu belassen. Der IHE Cross-reference Consumer bekommt seine

Informationen ausschließlich, indem er Anfragen an den CRM stellt und daraufhin Mengen von Referenz-ID's erhält. Zusätzlich gibt es noch die Möglichkeit diesen Vorgang mit einer IHE Update Notification zu verbinden. In diesem Fall wird die Anfrage nur noch für Situationen benötigt in denen der CRM und der IHE Cross-reference Consumer nicht miteinander synchron sind.

Das PIX Integrationsprofil realisiert die Integration unterschiedlicher IHE Patient Identifier Domains indem es eine Querverweisliste zwischen den verschiedenen ID's eines einzelnen Patienten in den unterschiedlichen Domänen nutzt. Das Konzept eines MPI wird meist als Einführung einer 'Master Patient Identity Domain' angesehen, welche dann hierarchisch den einzelnen teilnehmenden Domänen übergeordnet ist. Diese Situation kann man nun als Sonderfall betrachten, in dem alle ID's von Patienten einzelner Domänen immer mit der ID der Master Patient Identity Domain verknüpft werden. Im Prinzip kann im Rahmen einer PIX Implementierung diese Master Domain als weitere teilnehmende Domäne betrachtet werden die mit allen anderen Domänen in Verbindung steht. Es ist allerdings nicht nötig diese Master Domain hierarchisch höher zu stellen, das Integrationsprofil bietet auch die Möglichkeit lediglich einen föderalistischen Verbund von Domänen anzulegen. Das zentrale Element bleibt aber auch in diesem Fall der CRM der nicht pro Domäne, sondern lediglich einmal in dem ganzen Verbund vorhanden ist.

### 4.1.3. Transaktionen

Zur Realisierung seiner Funktionalität bedient sich PIX im Wesentlichen dreier unterschiedlicher Transaktionen, auf die ich hier kurz eingehen möchte. Als Format der beteiligten Nachrichten wurde HL7v2 gewählt. Im Fall der IHE Identity Feeds Version 2.3.1, in den beiden anderen Fällen Version 2.5. In der Abbildung 4.2 werden die teilnehmenden Aktoren an den jeweiligen Transaktionen dargestellt.

#### Patient Identity Feed

Der IHE Patient Identity Feed (ITI-8) ist die Transaktion, mit der Patienteninformationen und Identifikationsmerkmale einer neu angelegten Patienten-ID von einer IHE Patient Identity Source einer der teilnehmenden Domänen zum CRM übertragen werden. Nicht nur bei neu angelegten ID's, sondern auch bei der Zusammenfassung bestehender ID's eines Patienten, bei der Änderung der ID - oder deren zugrundeliegender Identifikationsmerkmale - kann dies mittels IHE Patient Identity Feed zum CRM propagiert werden. Das Integrationsprofil [ACC07b] unterscheidet innerhalb der ITI-8 Transaktionen zwischen fünf Typen, die abhängig von der Aktion die gerade vollzogen wird festgelegt sind.

- Aufnahme eines stationären Patienten

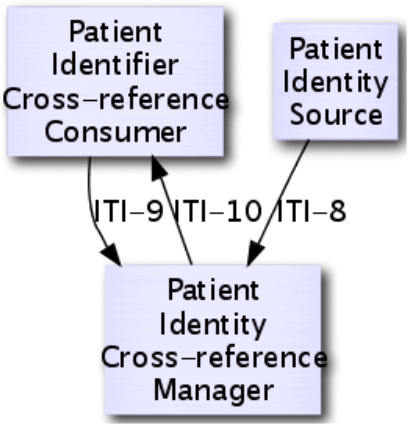


Abbildung 4.2.: PIX - Transaktionen

- Registrierung eines ambulanten Patienten
- Registrierung eines stationären Patienten (noch vor der Aufnahme)
- Änderung eines Datensatzes
- Zusammenfassen mehrerer Datensätze eines Patienten

Die ersten vier Typen dieser Aufzählung haben das gleiche Format. Sie enthalten zumindest die Segmente Message Header (MSH), Event Type (EVN), Patient Identification (PID) und Patient Visit (PV1). Alle weiteren Segmente aus dem HL7 Standard können zwar zusätzlich enthalten sein, sind aber optional. Diese vier Typen können anhand des Eintrags in der zweiten Komponente des MSH-9 Feldes der HL7 Nachricht unterschieden werden. Im PID Segment der Nachricht wird im Feld PID-3 die einzigartige ID des betroffenen Patienten übertragen, wobei hier die erste Komponente für die ID des Patienten genutzt wird, und in der vierten Komponente des Feldes die ID der Domäne übertragen wird. Bei dieser Transaktion wird von der IHE Patient Identity Source kein Attribut verlangt das über die im HL7 Standard definierten hinaus geht. Vom CRM wird erwartet, dass er zumindest ein bestimmtes Set von Attributen speichern kann welches benötigt wird, um zu Gewährleisten, dass es dem CRM möglich ist die Querverweise zu Datensätzen aus anderen Domänen zu generieren. [Tabelle A.1]

Sollte die ID der Domäne nicht in der Nachricht enthalten sein, so ist es dem CRM möglich dieses Feld auszufüllen, da hier als Konfigurationsdaten die Domänen-ID's aller teilnehmenden IHE Patient Identifier Domains hinterlegt sind, sowie zu jeder Domäne das als IHE Patient Identity Source berechtigte System. Die Identifikation des betreffenden Systems kann bespielsweise über IP-Adressen erfolgen.

Anhand der festgelegten Auswahl von Attributen die der CRM mindestens erfassen muss, lässt sich eingrenzen wie der Abgleich, nach dem gegebenenfalls die Querverweise gebildet werden, der Datensätze aus unterschiedlichen Domänen erfolgt. Genauere Informationen zu diesem Vorgang sind jedoch nicht in dem Anwendungsprofil hinterlegt[ACC07b]:

> „The cross-referencing process (algorithm, human decisions, etc.) is performed within the Patient Identifier Cross-reference Manager Actor, but its specification is beyond the scope of IHE."

Nach Abschluss des Abgleichs und der Erstellung der neuen Querverweise können diese mittels PIX Query beim CRM abgefragt werden. Sind teilnehmende IHE Patient Identity Consumer dementsprechend konfiguriert, werden diese auch direkt nach Erzeugung der neuen Querverweise hiervon mittels IHE PIX Update Notification in Kenntnis gesetzt.

Ein leicht angepasstes Format haben die Nachrichten des fünften Typs, welche verwendet werden wenn eine IHE Patient Identity Source feststellt, dass für einen Patienten zwei Datensätze innerhalb ihrer eigenen Domäne existieren, und diese daher zusammengefasst werden sollen. Die ersten drei Segmente dieser Nachricht entsprechen den obigen, also MSH, EVN und PID, jedoch wird an vierter Stelle das Segment Merge Information (MRG) eingefügt und das PV1 Segment ist in diesem Falle optional. Dabei werden die ersten drei Segmente exakt wie bei den anderen Typen besetzt, wobei im PID Segment - und falls vorhanden auch im PV1 - derjenige Datensatz verwendet wird, der nach der Verschmelzung übrig bleiben soll. Im MRG Segment wird die ID des Datensatzes übertragen, der nach der Verschmelzung nicht mehr verwendet werden soll. Es ist hierbei nicht zwingend erforderlich das dieser Datensatz gelöscht wird, er soll lediglich nach Abschluss der Aktion nicht mehr referenziert werden. Der CRM wird nach Eingang einer solchen Nachricht alle Referenzen die aktuell auf die im MRG Segment übertragene ID zeigen durch Referenzen auf die im PID Segment enthaltene ID ersetzen. Danach laufen die Arbeitsschritte wie bei Empfang einer ITI-8 Nachricht eines der zuvor beschriebenen Typen.

### PIX Query

Die Transaktion PIX Query (ITI-9) findet zwischen einem IHE Patient Identity Cross-reference Consumer und dem CRM statt. Der Consumer übermittelt eine ihm bekannte ID aus seiner Domäne zu einem Patienten und bekommt vom CRM eine Liste mit ID's des betroffenen Patienten in anderen teilnehmenden Domänen zurück, wie in Abbildung 4.3[ACC07b] dargestellt. Der IHE Cross-reference Consumer generiert eine HL7v2 Nachricht die zumindest aus den drei Segmenten MSH, Query Parameter Definition (QPD) und Response Control Parameter (RCP) besteht. Wie bereits bei

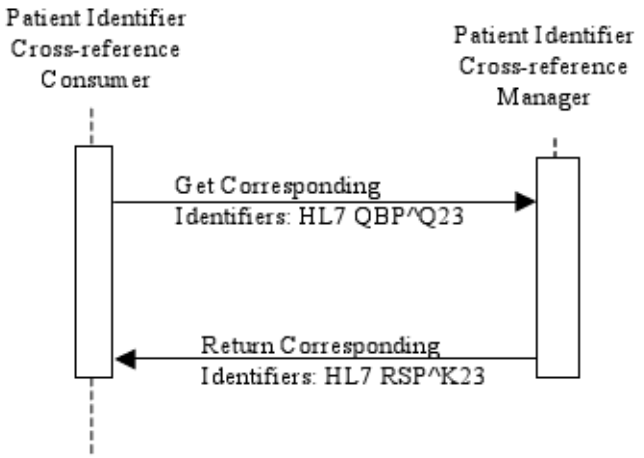


Abbildung 4.3.: PIX Query

den ITI-8 Nachrichten wird auch hier im MSH-9 Feld der Typ der Nachricht definiert. Im QPD-3 Feld wird die bekannte ID des Patienten übertragen, wobei dies im selben Format wie bei dem PID-3 Feld im IHE Identity Feed geschieht, also sowohl die lokale Patienten ID als auch die ID der Domäne enthalten ist. Es besteht weiterhin die Möglichkeit für den IHE Cross-reference Consumer in das Feld QPD-4 einzutragen, wenn nur Querverweise aus bestimmten Domänen angefordert werden sollen, also nicht alle teilnehmenden Domänen der IHE Cross-reference Domain für den IHE Cross-reference Consumer interessant sind. Das RCP Segment das in jeder dieser Nachrichten enthalten ist, muss nicht zwingend Informationen enthalten da das PIX Integrationsprofil keine Attribute hieraus benötigt, es ist allerdings im HL7 Standard festgelegt das nach einem QPD immer ein RCP Segment folgen muss daher ist dieses Segment nicht optional.

Wenn der CRM eine Nachricht dieses Typs empfangen hat, wird eine Antwortnachricht generiert die mindestens aus den Segmenten MSH, Message Acknowledgement (MSA), Query Acknowledgement (QAK) und QPD besteht. Falls es zu dem gestellten Query mindestens einen Querverweis gibt, so enthält diese Nachricht außerdem ein PID Segment. Wenn ein Query entweder eine dem CRM nicht bekannte Patienten ID enthält oder die ID der Domäne nicht bekannt ist, wird ein Error Segment (ERR) mit der entsprechenden Fehlercodierung in die Nachricht aufgenommen. Sollten unbekannte ID's bei den zu durchsuchenden Domänen vorkommen, so wird ein ERR Segment pro unbekannter ID erzeugt. In all diesen Fällen wird sowohl im MSA als auch im QAK Segment lediglich ein Fehlercode versendet also auch teilweise zu bearbeitende Anfragen werden bei Auftreten eines Fehlers komplett unbearbeitet zurück gesendet. Ist ein Query fehlerfrei zu bearbeiten, so wird im MSA-1 Feld ein 'application accept' gesendet, und im QAK-2 Feld wird festgehalten ob Datensätze

den angefragten Patienten betreffend gefunden wurden. Sofern dies zutrifft wird im dann vorhandenen PID Segment lediglich das PID-3 Feld gefüllt, und darin für jeden gültigen Querverweis innerhalb des Suchraumes das Datenpaar aus lokaler ID des Patienten und ID der zugehörigen Domäne gespeichert. Sollte es in einer der betroffenen Domänen mehrere ID's zu diesem Patienten geben, so wird der CRM diese sukzessiv innerhalb der Ergebnisliste übertragen. Vom IHE Cross-reference Consumer wird erwartet, dass entweder all diese multiplen ID's der Fremddomäne verwendet oder falls ein IHE Cross-reference Consumer nicht in der Lage ist mit multiplen ID's zu arbeiten alle ignoriert werden. Dadurch soll verhindert werden das unvollständige Daten vom IHE Cross-reference Consumer präsentiert werden.

### PIX Update Notification

Wie bereits der Query findet die IHE PIX Update Notification (ITI-10) ebenfalls unter Beteiligung des IHE Cross-reference Consumers und des CRM statt. Wie Eingangs des Kapitels über PIX erwähnt, müssen eine Reihe von administrativen Prozessen festgelegt sein die nicht Inhalt des Integrationsprofils sind, aber benötigt werden um dessen Funktionalität zu gewährleisten. Dazu gehört unter anderem ein Prozess, über den es einem IHE Cross-reference Consumer möglich ist sich beim CRM für eine Reihe von Domänen einzutragen, sodass der IHE Cross-reference Consumer benachrichtigt wird sollte sich eine Änderung an den Querverweisen in einer der ausgewählten Domänen ergeben. Hierbei wird die Konfiguration welche Consumer über welche Domänen informiert werden sollen im CRM gespeichert.

Die erforderlichen Segmente für eine ITI-10 Nachricht sind MSH, EVN, PID und PV1. Im MSH-9 Feld wird auch hier der Typ der Nachricht angezeigt. Die EVN und PID Segmente werden in der gleichen Weise genutzt wie bei der Antwortnachricht auf ein PIX Query, und im PV1 Segment wird lediglich ein Vermerk untergebracht das es sich nicht um Patienteninformationen eine aktuelle Behandlung betreffend handelt. Ein empfangender IHE Cross-reference Consumer behandelt die Nachricht ebenso wie eine Antwort auf ein gestelltes Query.

## 4.2. PIDS

Das Ziel der OMG Person Identification Service Spezifikation ist es, ein ID Management für Personen zu realisieren, welches speziell auf die Bedürfnisse im Gesundheitsbereich abgestimmt ist. Um dies zu erreichen werden sowohl die Vergabe von ID's innerhalb einer Domäne, als auch der Bezug auf ID's anderer Domänen unterstützt. Außerdem soll die Suche und die Bildung von Querverweisen, sowohl durch User als auch durch automatisierte nachrichtengesteuerte Algorithmen, möglich sein. Der Zusammenschluss von OMG Person Identification Services soll topologisch unabhängig

geschehen können. Es soll den PIDS Implementierungen unter den verschiedensten Sicherheitsmechanismen und Vertraulichkeitsregelungen möglich sein die nötige Datensicherheit und die Datenvertraulichkeit zu gewährleisten. Die Kompatibilität der unterschiedlichen Implementierungen soll dadurch erreicht werden, dass es eine minimale Anforderung gibt welche Elemente vorhanden sein müssen. Die spezifischen Anforderungen an eine Implementierung in einem bestimmten Umfeld wird durch zusätzliche Elemente oder die Konfiguration der Standardelemente erzielt. Des Weiteren dient der Service dazu, für verschiedene Größenordnungen die nötigen Ordnungsgrade zu definieren, von Query-only Systemen mit zentralisiertem Speicher der ID's bis hin zu föderativen Systemen bestehend aus mehreren ID-Domänen.

### 4.2.1. Aufbau

Der grundlegende Baustein des PIDS Modells ist eine ID Domäne. Ähnlich wie schon beim PIX gilt in jeder dieser Domänen, dass eine Person über eine einzigartige ID eindeutig zu bestimmen ist. Idealerweise existiert innerhalb einer Domäne genau eine ID für jede Person, es können aber auch beliebig viele Duplikate vorhanden sein, so dass mehrere ID's auf eine Person zeigen. Eine solche Domäne kann bereits aus mehreren Systemen bestehen die gemeinsame ID's benutzen, wie beispielsweise einem bestehenden MPI und seiner Teilnehmer. Mehrere dieser ID Domänen können nun zu einer OMG Correlating ID Domain zusammengefasst werden. Des Weiteren kann eine solche wiederum als ID Domäne an einem größeren Verbund teilnehmen, also eine Baumstruktur aufgebaut werden, wie in Abbildung 4.4[Obj01] dargestellt. Da es für eine einzelne ID Domäne auch möglich ist an verschiedenen OMG Correlating ID Domains teilzunehmen lassen sich aber auch Peer-Strukturen aufbauen.

Die OMG Correlating ID Domain ist in diesem Modell das koordinierende Strukturelement, nur über diese ist der Zugriff auf die korrelierenden Daten der teilnehmenden Domänen möglich. Hier wird auch das Framework bereitgestellt über das die Generierung der Querverweise, übergreifend über die einzelnen ID Domänen, erfolgt. Um die Funktionalität des Modells zu realisieren, werden Schnittstellen für jedes Element definiert über die alle Zugriffe auf dessen interne Daten erfolgen. Dadurch erleichtert sich auch die Zugriffskontrolle, da nur in diesen Schnittstellen eine Rechteverwaltung und eine Authentifizierung nötig ist, und im Fall einer externen Anfrage die weitere Bearbeitung innerhalb der Domäne gleich behandelt werden kann. Jede teilnehmende Domäne kann hierbei selbständig entscheiden, welche Anfragen abhängig von Anfragesteller und Art der angefragten Daten genehmigt werden soll.

Das PIDS Modul ist mit Hilfe der OMG Common Object Request Broker Architecture zu realisieren, deren Ziel es ist verschiedenen Applikationen zu ermöglichen miteinander zu kommunizieren ohne große interne Anpassungen nötig zu machen. Dies wird mit Hilfe von OMG Object Request Brokern erledigt. Die dazu nötigen Definitionen der Schnittstellen erfolgen in der ebenfalls von der OMG definierten OMG

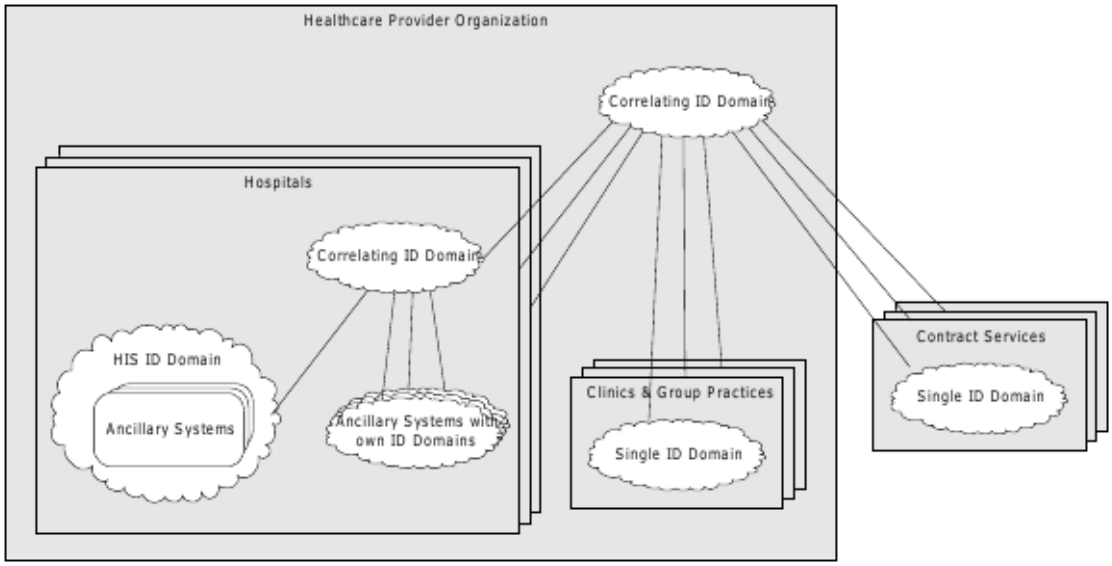


Abbildung 4.4.: Domain Reference Model for PIDS

Interface Definition Language (IDL).

### 4.2.2. Interfaces

Auf die verwendeten Schnittstellen im PIDS Modell möchte ich hier kurz eingehen, diese sind abgeleitet vom OMG IdentificationComponent Interface. Dieses referenziert wiederum die anderen Interfaces wie in Abbildung 4.5[Obj01] dargestellt. Eine OMG IdentificationComponent hat eine Reihe optionaler Interfaces die sie implementieren kann. Auf diese Art wird gewährleistet, dass die Funktionalität an die Bedürfnisse des jeweiligen Umfeldes, beziehungsweise der Applikationen, angepasst werden kann. Es ist sowohl möglich eine solche Komponente durch ein einziges Objekt zu implementieren, indem alle benötigten Interfaces in ein spezifisches Interface übernommen werden, als auch für jedes unterstützte Interface ein eigenes Objekt zu generieren wodurch diese getrennt voneinander behandelt werden können. Falls man allerdings mehr als ein Objekt nutzt um die Komponente umzusetzen, muss aus Konsistenzgründen darauf geachtet werden, dass alle Attribute von den Objekten identisch behandelt werden, da die Applikation die Komponente als ganzes wahrnimmt. Es müssen also alle Objekte auf einen gemeinsamen Datenbestand zugreifen, oder falls das nicht möglich ist, muss durch die spezifische Implementierung gewährleistet sein das keine Inkonsistenzen auftreten. Eine solche OMG IdentificationComponent kann an unterschiedlichen Stellen eingesetzt werden. In einem kleinen Hilfssystem müssen hierbei weniger Interfaces implementiert werden als wenn man die Komponente als

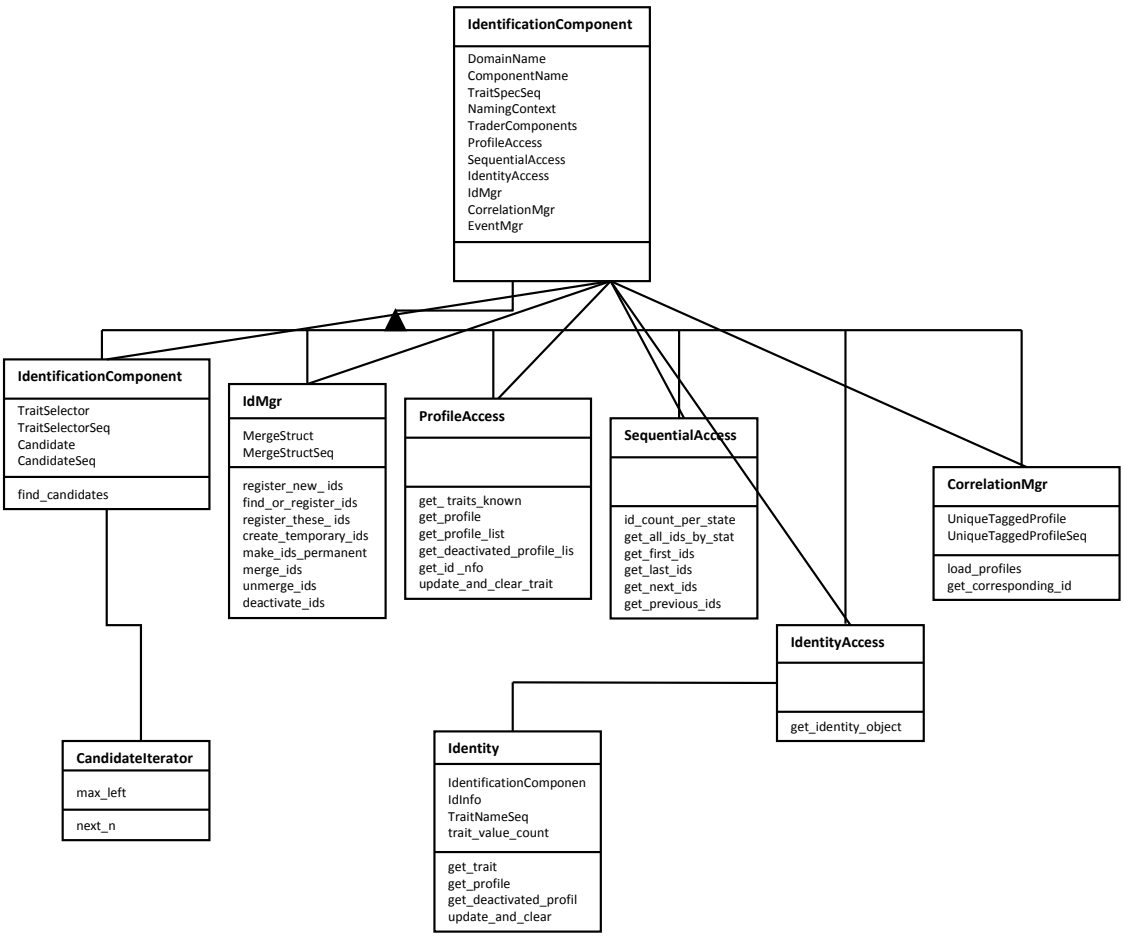


Abbildung 4.5.: PIDS Components and Inheritence Diagram

vollständiges MPI System nutzt.

Die charakterisierenden Informationen zu einer Person werden in PIDS in sogenannten 'Traits' gespeichert:

```
typedef NamingAuthority::QualifiedNameStr TraitName;
typedef any TraitValue;
struct Trait {
TraitName name;
TraitValue value;
};
```

Es existiert ein Modul OMG HL7Version2_3 in dem die Traitnamen für die Verwendung des HL7 Standards definiert werden. Hierbei werden die Namen der Felder 5-30 des PID Segmentes als Strings verwendet. Außerdem werden die Datentypen soweit

im HL7 Standard definiert übernommen. Es wird an dieser Stelle keine Auswahl getroffen, anhand welcher Attribute eine Bildung von Querverweisen später erfolgen kann. Diese Entscheidung wird allein der spezifischen Implementierung überlassen.

> Many of these traits from the HL7 PID segment are demographic traits instead of identifying traits. The difference between demographic traits and identifying traits is often dependent on the environment in which they are being used. It was decided to include all of these PID segment traits and let the PIDS implementors choose the ones that are considered identifying traits for their service.[Obj01]

### ProfileAccess

Das Interface OMG ProfileAccess ist die grundlegende Schnittstelle über die innerhalb des PIDS Traits zu Personen abgerufen werden können. Zum einen existiert eine Methode durch die abgerufen werden kann welche Traits zum angefragten Datensatz verfügbar sind, zum anderen gibt es die Möglichkeit sich die Profile einer oder mehrerer Personen durch die Übergabe der ID's ausgeben zu lassen. Sollten Änderungen an bereits bestehenden Datensätzen vorzunehmen sein haben diese ebenfalls über die ProfileAccess Schnittstelle zu erfolgen, wobei die Überprüfung auf Schreibrechte auch an dieser Stelle erfolgt.

### SequentialAccess

Mit Hilfe des OMG SequentialAccess Interfaces können, durch Übergabe einer Reihe von Parametern, Filterfunktionen ausgeführt werden, wobei je nach aufgerufener Methode die Antwort von der reinen Anzahl der Ergebnisse bis hin zu einer Sequenz von Profilen, die bereits auf in der Anfrage übergebene Traits projiziert wurde, reicht. Im Falle einer Anfrage die eine Sequenz von Profilen zurück liefern soll kann auch ein bestimmtes Fenster in der Sequenz, definiert in Größe und Position, angefragt werden.

### IdentityAccess

Innerhalb des OMG IdentityAccess Interfaces ist ein weiteres Interface definiert welches OMG Identity heißt. Die OMG Identity-Objekte die hier erzeugt werden, können nur vom OMG IdentityAccess Interface gelesen werden und dienen in erster Linie der Zugriffskontrolle. Dabei ist es möglich die Rechtevergabe für die einzelnen ID's unabhängig voneinander zu steuern.

### IdentifyPerson

Durch dieses Interface wird die Funktionalität realisiert, dass eine Anfrage, in der einige Traits und deren Werte übergeben werden, an eine ID Domäne gestellt werden kann, deren Ziel es ist alle möglichen Datensätze, die sich auf die gleiche Person beziehen, zurück zu erhalten. Beim Stellen der Anfrage übergibt man einen Parameter, welcher die Stärke der nötigen Übereinstimmung festlegt. In der Rückgabemenge befinden sich alle möglichen Kandidaten denen mindestens dieser Wert zugeordnet wurde. Was den Aufruf der 'Matching Engine' angeht so sind noch verschiedene Parameter vorgesehen, wie die Gewichtung einzelner Traits. Da in der Spezifikation keine Engine festgelegt wird ist es implementationsabhängig ob diese Faktoren beachtet oder ignoriert werden.

### IdMgr

Innerhalb von PIDS kann jede ID eine bestimmte Menge von Zuständen annehmen. Im Einzelnen sind das: Temporary, Permanent, Invalid, Deactivated und Unknown. Diese Zustände werden von den bisher angeführten Interfaces bei allen Aktionen als eine Art Filter beachtet. Im OMG IdMgr werden die nötigen Funktionen zur Verfügung gestellt, mit Hilfe derer man die ID's in einer einzelnen Domäne verwalten kann. Dies betrifft sowohl Neuerstellung von ID's, als auch Deaktivierungen und Verschmelzungen, falls einer Person innerhalb der Domäne mehrere ID's zugewiesen wurden. Sollte das Interface getrennt von der ID generierenden Stelle der Domäne betrieben werden, so ist es ebenfalls möglich mit einem Profil die gewünschte ID zu übergeben, wobei lediglich die Verknüpfung dieser hier erfolgt.

### CorrelationMgr

Diese Schnittstelle wird in einer teilnehmenden ID Domäne nicht umgesetzt, sie ist lediglich in der OMG Correlating ID Domain implementiert. Es ist möglich zu einer übergebenen ID alle existierenden Querverweise innerhalb der OMG Correlating ID Domain zu erhalten. Allerdings wird keine Möglichkeit zur Verfügung gestellt direkt eine Verknüpfung von Profilen unterschiedlicher teilnehmender Domänen herzustellen, es können lediglich Profile in die OMG Correlating ID Domain geladen werden. Wann allerdings die Vergleichoperationen und gegebenenfalls die Bildung der Querverweise erfolgt ist eine Entscheidung bei der Implementierung und nicht durch Aufrufe beeinflussbar. Optional kann an dieser Stelle auch eine Methode implementiert werden, durch die beim Laden von Profilen in die OMG Correlating ID Domain für jedes dieser Profile eine ID zurückgeliefert wird, wodurch übergreifend über die einzelnen ID Domänen einzigartige ID's entstehen. Dies ist bespielsweise bei der Nutzung eines PIDS Systems als MPI nützlich, es können dadurch aber auch bei

| Conformance Class | Identify Person | Profile Access | Sequential Access | ID Mgr. | Identity Access | Correlation Mgr. |
|---|---|---|---|---|---|---|
| Simple PIDS | * | * | | | | |
| Sequential Access PIDS | * | * | * | | | |
| ID Domain Mgr PIDS | * | * | | * | | |
| Identity Access PIDS | * | | | | * | |
| Correlation PIDS | | | | | | * |

Tabelle 4.1.: Conformance Classes

hierarchischen Strukturen Anpassungen am Format getroffen werden, falls dies für Repräsentation nach außen erforderlich ist.

### 4.2.3. ConformanceClasses

In der Spezifikation von PIDS ist eine Reihe von OMG Conformance Classes angegeben. Diese sollen als Referenzen dienen, anhand derer eine implementierte Schnittstelle klassifiziert werden kann. Um die Bedingungen einer der OMG Conformance Classes zu erfüllen gibt es eine Reihe von allgemeinen Grundvorgaben. So müssen unter anderem die semantischen Vorgaben für eine OMG IdentificationComponent, wie die Konsistenz aller beinhalteten Interfaces bezüglich der Anwendung, erfüllt sein. Es muss mindestens ein Trait unterstützt werden, und falls die Traits aus beispielsweise dem OMG HL7Version2_3 Modul verwendet werden, so müssen auch die anderen in diesem Modul definierten semantischen Vorgaben (Datentypen etc.) eingehalten werden.

Die einzelnen OMG Conformance Classes stehen für eine gewisse Funktionalität, welche gewährleistet ist, sofern eine Implementierung die Anforderungen einer solchen erfüllt. Durch eine OMG Simple PIDS werden die nötigen Operationen, zum Erhalten eines Profils mit Hilfe einer einer ID, und zum Vergleich mehrerer ID's unter Berücksichtigung einiger Traits zur Verfügung gestellt. Durch ein OMG Sequential Access PIDS wird zusätzlich zu diesen noch die Möglichkeit geboten, durch eine Menge von ID's sequentiell zu blättern. Ein OMG ID Domain Mgr PIDS bietet zusätzlich zu den Funktionen eines OMG Simple PIDS noch die Möglichkeit ID's zu administrieren, also sowohl Erzeugungs- als auch Änderungs- und Löschoperationen. Als alternative Zugriffsstelle gilt ein OMG Identity Access PIDS, welches im Grunde die gleichen Funktionen bietet wie ein OMG Simple PIDS allerdings die OMG IdentityAccess Schnittstelle nutzt, wodurch ID abhängige Zugriffskontrollen, zum Beispiel

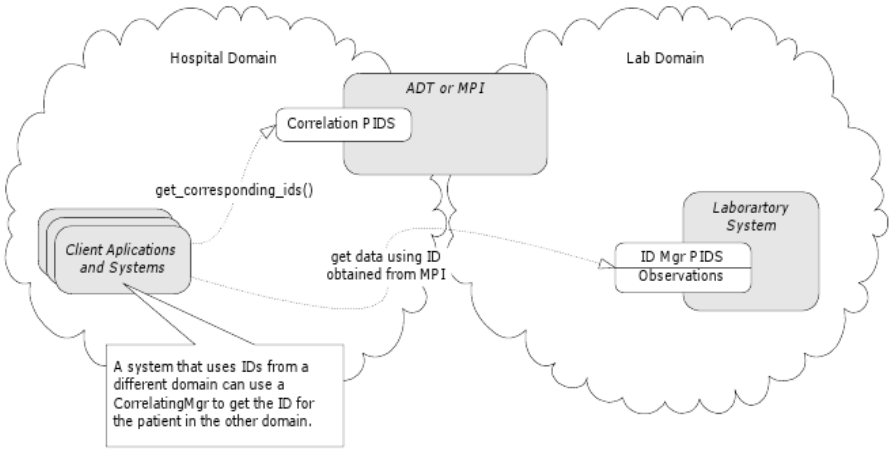


Abbildung 4.6.: MPI System mit PIDS Interface

falls ein Nutzer die Zugriffsrechte auf sein Profil selbst festlegen darf, ermöglicht werden. Die Klasse OMG Correlation PIDS wird erfüllt, falls die Schnittstelle OMG CorrelationMgr. implementiert ist und zumindest die drei Traits: HL7:PatientName, HL7:DateTimeOfBirth und HL7:Sex in der Liste der unterstützten Traits stehen. Es ist für eine Implementierung möglich mehrerer dieser Klassen anzugehören, wenn jeweils alle Anforderungen der einzelnen Klassen erfüllt sind. In Abbildung 4.6[Obj01] ist eine grafische Darstellung wie ein solches System in Verbindung mit einem MPI genutzt werden kann.

## 4.3. MPI Implementierungen

Im Folgenden Abschnitt werden exemplarisch zwei bestehende Implementierungen von MPI Systemen betrachtet. Der promedtheus MPI ist ein kommerzielles Produkt, welches bereits in eineigen Kliniken in Deutschland im Einsatz ist. Der MPI von Sun ist ein konfigurierbares Repository als Teil der Java Composite Application Plattform Suite. Diese beiden Umsetzungen werden nun im Einzelnen dargestellt.

### 4.3.1. promedtheus MPI

Zu den grundlegenden Anforderungen an einen MPI und dessen Funktionsumfang, sowie dem Lösungsansatz des promedtheus MPI zu diesen Aufgaben, schreibt Herr Wünnemann in einem Artikel [WD01] von 2001 eine kurze Zusammenfassung. Hier wird darauf eingegangen, dass es im medizinischen Umfeld so bald Informationssysteme einen gewissen Umfang erreichen unvermeidbar ist, mit einer Heterogenität der zugrundeliegenden Systeme umzugehen. Als Teilschritt wird hier die Einführung von

Kommunikationsservern zum Austausch von HL7 standardisierten Nachrichtenformaten genannt, jedoch ist dies nicht ausreichend um eine wirkliche Integration zu gewährleisten. Es ist nötig Softwarekomponenten zu entwickeln, welche als Schnittstelle zwischen den bestehenden Systemen dienen können und mit Hilfe derer es auch möglich ist, auf Veränderungen der organisatorischen Bedingungen zu reagieren. Im Zuge der Verknüpfung der unterschiedlichen Anwendungen werden auch Single-Sign-On Systeme genannt, die eine Erleichterung für den Benutzer darstellen sollen und außerdem ermöglichen verschiedene Anwendungen auf einen Patienten zu synchronisieren. Auch werden im medizinischen Umfeld hohe Anforderungen an die Stabilität der verwendeten Informationssysteme gestellt sowie spezielle Sicherheitsanforderungen die eingehalten werden müssen.

Als Erweiterung der normalen Aufgaben eines MPI wird neben einigen anderen auch die Re-Identifikation von Patienten aufgeführt. Da es zum effektiven Aufbau einer elektronischen Patientenakte über einen längeren Zeitraum, um Zugriff auf wichtige patientenspezifische Besonderheiten (Allergien, Vorbefunde) zu haben, nötig ist

> „wiederkehrende Patienten trotz geänderter Daten [...] auch über einen längeren Zeitraum als solche zu erkennen“. [WD01]

Betont wird hier die Wichtigkeit der erneuten Zuweisung einer bereits vorhandenen Patienten-ID.

Um einen MPI in diesem Zusammenhang nutzen zu können, ist es ideal wenn die Rolle des patientenführenden Systems dem MPI übertragen wird, und die ADT-Funktionen (Admission-Discharge-Transfer) aus den einzelnen Komponenten entnommen und von einem eigenständigen Referenzsystem aus betrieben werden. Um den MPI in allen Systemen eines KIS zum Einsatz zu bringen bietet sich, aufgrund der niedrigen Anforderungen an die einzelnen Teilnehmer, die gute Integrierbarkeit in bestehende Systeme und die einfache Wartung eine web-basierte Oberfläche an. Als Erweiterungsstrategie um MPI Systeme organisationsübergreifend zu nutzen wird die Einführung von standortbasierten MPIs und die Replikation dieser untereinander erwähnt.

Als Lösungsansätze zu den zuvor aufgeführten Problemen werden die einzelnen Komponenten des promedtheus MPI, welcher sich nach erfolgreicher Pilotphase bereits seit 1.8.2000 am Klinikum Memmingen im Betrieb befindet[Mem01], und deren Funktionen beschrieben. Die kompletten ADT-Transaktionen können abgewickelt werden, es gibt die Möglichkeit alle intern im XML-Format gespeicherten Daten in HL7 Nachrichten abzubilden, und die angesprochene Funktionalität zur Replikation mehrerer MPI Systeme in einem Verbund ist bereits inbegriffen. Ein dem CCOW-Standard entsprechender Kontextmanager ermöglicht die Synchronisation der verschiedenen Anwendungen an einem Arbeitsplatz auf einen Patienten hin. Bezüglich Identifikation steht ein ID-Server zur Verfügung, der alle Objekte mit einzigartigen ID's versieht. Diese ID's sind, ihr spezielles Objekt betreffend, semantik-

frei, um später erforderlichen Korrekturen vorzubeugen. Über eine Kontingentverwaltung soll gesichert werden, dass auch bei aktuell nicht verfügbarer Verbindung zum Server keine Konflikte auftreten. Im Zuge aller Aufnahmevorgänge wird der Re-Identifikationsprozess gestartet der über hier nicht näher spezifizierte intelligente Algorithmen nach Ähnlichkeitsmerkmalen zu bestehenden Patienten sucht, welche weit über das klassische Name, Vorname und Geburtsdatum hinaus gehen. Dadurch soll gewährleistet werden, dass auch Patienten mit mittlerweile geänderten oder in der Vergangenheit falsch aufgenommenen Daten erkannt werden.

> „Besondere Suchmechanismen ermöglichen die Re-Identifizierung von Patienten über den reinen Vergleich von Name, Vorname und Geburtsdatum hinaus, so dass ein für die medizinische Dokumentation und den Aufbau elektronischer Archive unbedingt erforderliche konsistente Patientendatenbestand entsteht.“ [Mem01]

Da es sich bei dem promedtheus MPI um ein kommerziell vertriebenes Produkt handelt, werden die genaueren Algorithmen und Vorgänge innerhalb der einzelnen Komponenten nicht komplett offen gelegt. Die elektronischen Patientenakten werden nicht komplett in dem MPI gespeichert, sondern lediglich die zum Betrieb des MPI erforderlichen Daten werden in den zentralen Bestand übernommen. Hierbei werden auch rechtliche Grenzen bezüglich Datenschutz und Datensicherheit angesprochen, die im jeweiligen Einsatzgebiet zu klären sind.

### 4.3.2. Sun MPI

Im Rahmen der Java Composite Application Plattform Suite (CAPS) von Sun existiert ein Repository zur Umsetzung von Sun Master Indexes [Sun08c]. Auf dieser Plattform gibt es mit dem Sun Master Patient Index ein eMPI System. Ein Sun MPI wird mit Hilfe von NetBeans innerhalb eines Projektes implementiert. Ein solches Projekt besteht aus den Konfigurationsdateien der genutzten Datenbank und anderen Java CAPS Komponenten, mit deren Hilfe es möglich ist die Implementierung an die spezifischen Anforderungen anzupassen. Wenn von einem Teilsystem ein neuer Datensatz an den MPI gesendet wird, so wird im Falle eines neuen Patienten für diesen eine, für den gesamten Bereich gültige, ID erzeugt. Im MPI wird der eingegangene Datensatz zusammen mit der lokalen ID des Patienten im Ursprungssystem und der ID des Ursprungssystems selbst als 'system record' gespeichert. Ein solcher Datensatz ist immer Teil eines 'enterprise record', welcher pro Patient nur genau einmal existiert. In diesem werden alle dem Patienten zugeordneten Datensätze die beim MPI ankommen als einzelne 'system records' gespeichert, sowie ein 'single-best-record' (SBR). Für den SBR werden alle zugrunde liegenden Datensätze untersucht, und für jedes Attribut das in mehreren Datensätzen vorhanden ist, wird anhand mehrerer

Parameter, wie zum Beispiel der Zuverlässigkeit der Datenquelle oder Zeitstempel des Eintrages, entschieden welcher Eintrag die zuverlässigsten Daten bietet.

Bevor die Daten im MPI gespeichert werden wird eine Reihe von Operation auf ihnen durchgeführt, so werden zum Beispiel Adress- oder Personendaten zuerst normalisiert. Im Falle der Addressdaten werden diese in der Standardkonfiguration in folgende vier Komponenten zerlegt:

- House number, rural route identifier, P.O. box number (Address.HouseNumber)
- Match street name, rural route descriptor,
  P.O. box descriptor (Address.StreetName)
- Street direction (Address.StreetDir)
- Street type (Address.StreetType)

Nach Abschluss dieses Vorgangs werden einige Attribute, darunter auch der Straßenname, phonetisch kodiert und in eigene Komponenten gespeichert. Im Falle der Personendaten sind dies acht Attribute, wodurch die spätere Suche nach Datensätzen erleichtert werden soll. Es stehen auf den fertigen Datensätzen verschiedene Typen von Suchanfragen zur Verfügung. Hierbei handelt es sich um eine einfache, sowie eine phonetische Version einer alphanumerische Suche. Außerdem existieren noch Bereichsanfragen die unter anderem dazu eingesetzt werden, Dubletten zu erkennen und entsprechend bearbeiten zu können.

> Sun Master Patient Index uses the Sun Match Engine, a proprietary algorithm for probabilistic matching of patient records
> and data standardization.[Sun08a]

Zur Generierung der phonetischen Kodierungen der betroffenen Attribute stellt die Sun Match Engine bereits einige Algorithmen zur Verfügung. Darunter ist der Soundex Algorithmus sowie Erweiterungen von diesem, zwei Variante des Metaphone und der NYSIIS Algorithmus.

In der Grundkonfiguration des Sun MPI setzt dieser genau folgende fünf Felder ein, um Patienten miteinander zu vergleichen:

- Person.StdFirstName
- Person.StdLastName
- Person.SSN
- Person.DOB
- Person.Gender

Zu diesem Vorgang lassen sich nun Schwellwerte einstellen, ab welchem Grad der Übereinstimmung Aktionen durchgeführt werden sollen. Ebenso ist es möglich die Felder auf denen dieser Abgleich stattfindet zu editieren.

Durch einen Sun MPI lässt sich durch passende Konfiguration die komplette Funktionalität eines IHE PIX - Cross-reference Manager realisieren. Hierdurch ist es möglich die Vorteile der Java Composite Application Platform Suite zu nutzen und gleichzeitig den Vorteil des IHE Integrationsprofils, dass an die einzelnen teilnehmenden Domänen keine erhöhten Anforderungen gestellt werden müssen, zu erhalten. [Sun08b]

## 4.4. Zusammenfassung

In diesem Kapitel wurden zwei verschiedene Frameworks sowie zwei bestehende Implementierungen zu MPI Systemen dargestellt. Beide Frameworks nutzen vorhandene und bereits relativ verbreitete HL7 Standards um die Kompatibilität zu möglichst vielen bestehenden Systemen zu erreichen. In beiden Fällen können die domäneninternen Kommunikationsmethoden unverändert weiter verwendet werden, lediglich für die domänenübergreifenden Prozesse sind die spezifizierten Methoden anzuwenden. Diese Kommunikation kann im Falle von PIX komplett durch HL7 Nachrichten erfüllt werden. PIDS kann zwar die Felder von HL7 Nachrichten nutzen, die Übertragungen laufen allerdings über CORBA Schnittstellen.

Das PIX Integrationsprofil versucht möglicht große Teile der Funktionalität im IHE Cross-reference Manager zu realisieren, um die Anforderungen an die einzelnen Teilnehmer zu reduzieren. Es gibt genaue administrative Voraussetzungen an die sich eine Einheit die als IHE Patient Identification Domain teilnimmt halten muss. Dabei wird aber nicht genau definiert nach welchem Algorithmus oder auf genau welchen Attributen die Vergleichsoperationen durchgeführt werden sollen. Lediglich eine festgelegte Mindestmenge an Attributen die beim Anlegen eines Patienten im CRM übertragen werden muss ist spezifiziert, diese ist in Tabelle 4.2 angezeigt. In diesem Zusammenhang ist auch eine Mindestmenge an Attributen definiert, die von der Implementierung des CRM gespeichert werden können muss [Anhang A.1] um genug identifizierende Daten für die Vergleichsoperationen zur Verfügung zu stellen. Auch auf Probleme, wie beispielsweise ein Tippfehler im Namen eines Patienten in einer der Domänen, wodurch ein 'fuzzy matching' nötig ist, wird im Rahmen der Profilspezifikation nicht eingegangen. Durch die starke Zentralisierung ist die Realisierung eines MPI relativ einfach gestaltet, indem man eine, möglicherweise hierarchisch höher gestellte, IHE Master Patient Identity Domain definiert die potentiell keine eigenen medizinischen Daten verwaltet, sondern lediglich mit allen anderen Domänen verknüpft ist und deren ID's vom CRM als Master-ID verwendet werden.

In der Spezifikation von PIDS werden verschiedene Arten von Schnittstellen de-

| Attribut | PIX | PIDS | Sun |
|---|---|---|---|
| Vorname | PatientName | HL7:PatientName | Person.StdFirstName |
| Nachname | PatientName | HL7:PatientName | Person.StdLastName |
| Social Security Number | SSN Number Patient | | Person.SSN |
| Geburtsdatum | Date of Birth | HL7:Date of Birth | Person.DOB |
| Geschlecht | Administrative Sex | HL7:Sex | Person.Gender |
| Mädchenname der Mutter | Mother Maiden's Name | | |
| Anschrift | Patient Address | | |
| Telefon - Heim | Phone Number Home | | |
| Telefon - Beruflich | Phone Number Business | | |
| Führerschein-nummer | Driver Licence Number - Patient | | |
| | | HL7-PID Feld 5-30 | |

Tabelle 4.2.: Attribute für Vergleichsoperationen

finiert. Durch Kombination dieser ist es möglich die an einer Stelle im System jeweils nötige Funktionalität zu gewährleisten. Im Bezug auf ein MPI System wird in PIDS lediglich ein Interface definiert, über das man die Ergebnisse einer Anfrage zu einem Patienten erhalten, beziehungsweise Patientendaten zur möglichen Verknüpfung bereitstellen kann. Auf die Implementierung der Vergleichsoperation wird auch in diesem Framework nicht näher eingegangen. Bei Nutzung des HL7 Moduls bei den identifizierenden Merkmalen beinhaltet die Schnittstelle eine große Menge an Attributen, um den Algorithmus der von der Implementierung genutzt wird flexibel wählen zu können. Die Konformitätsklasse gilt aber bereits bei den drei in der Tabelle angegebenen Attributen als erfüllt. Alternativ zu den Schnittstellen an denen Profile abgerufen werden können gibt es im Rahmen von PIDS noch OMG IdentityAccess Interfaces, bei denen die Zugriffsrechte nicht nur abhängig von dem aktuellen Benutzer oder System sind, sondern je nach ID einzeln konfiguriert werden können. Bei den beiden MPI Implementierungen die betrachtet wurden, handelt es sich beim promedtheusMPI um ein kommerzielles Produkt, daher sind in diesem Fall keine Information über die genaue Datenstruktur und Vergleichsoperationen angegeben. Es wird auf einige praktische Probleme eingegangen, wie beispielsweise das man durch Vereinigung des MPI mit dem ADT-System verhindern kann, dass ein

Patient bei erneuten Besuchen eine neue ID zugewiesen bekommt, die dann wieder mit Hilfe des MPI verknüpft werden muss. Ebenso werden die Vorteile bei der Verwaltung solcher Systeme mittels web-basierten Oberflächen und Kontextmanagern dargestellt. Bei der Sun MPI handelt es sich um ein Repository mit dessen Hilfe, unter anderem zu PIX konforme, MPI Systeme realisiert werden können. Im MPI werden nicht nur die Attribute gespeichert und für Vergleichsoperationen verwendet, sondern zur Erleichterung der weiteren Suchvorgänge wird aus jedem der Attribute, die in mehreren Datensätzen zu dem gleichen Patienten vorhanden sind, mittels verschiedener Parameter ein 'SingleBestRecord' gebildet. Dieser stellt die vollständigste und verlässlichste Repräsentation der identifizierenden Daten des Patienten im Gesamtsystem dar. Für die Vergleichsoperationen in dieser Implementierung wird die 'Sun Matching Engine' verwendet, deren in der Standardkonfiguration verwendeten Attribute sind in Tabelle 4.2 benannt.

# 5. Verwandte Arbeiten

Um die einen Patienten identifizierenden Daten im Kontext des verteilten MPI zu nutzen werden innerhalb des entworfenen Konzeptes eine Reihe von Manipulationen auf diesen Attributen durchgeführt. Darunter fallen Standardisierung, Phonetisierung und die Anwendung von Hashverfahren. Hier soll ein kurzer Überblick über die Funktionsweise einiger eingesetzter Techniken gegeben werden.

## 5.1. Phonetische Algorithmen

Unter der phonetischen Kodierung einer Zeichenkette versteht man im Allgemeinen die Abbildung auf einen (meist eingeschränkten) Wertebereich, wobei in ihrer Aussprache ähnliche, beziehungsweise identische Worte den gleichen Code ergeben sollen. Es existieren eine Reihe von Algorithmen mit deren Hilfe solche Kodierungen vorgenommen werden können. Der am weitesten verbreitete darunter ist der Soundex [Rep07] Algorithmus, der ursprünglich für den Klang von Wörtern in der englischen Sprache entwickelt wurde. Weitere nennenswerte Alternativen sind der Metaphone, beziehungsweise der Double Metaphone, der ebenfalls Wörter ihrem Klang in der englischen Sprache nach kodiert. Der NYSIIS Algorithmus ist exlizit auf Namen im englischsprachigen Raum angepasst. Speziell auf die deutsche Sprache hin ausgerichtet ist die sogenannte Kölner Phonetik. Diese Algorithmen werden im Folgenden kurz in ihrer Funktionsweise beschrieben.

### 5.1.1. Soundex

Um auf die Unterschiede der Funktionsweisen der einzelnen Algorithmen einzugehen werden diese nun im einzelnen kurz dargestellt. Soundex, als ältester dieser Algorithmen, wurde bereits 1918 patentiert. Er wird von anderen Methoden der phonetischen Verarbeitung von Wörtern meist als Referenz angegeben wenn es um die Leistungsfähigkeit der Ergebnisse geht. Wird Wort zur Kodierung an den Soundex Algorithmus übergeben, so bleibt der Anfangsbuchstabe unverändert, im Rest des Wortes werden alle Vokale, sowie die Konsonanten 'H', 'W' und 'Y' nicht mit kodiert. Alle anderen Konsonanten haben eine zugewiesene Ziffer, deren Sinn es ist ähnlich klingenden Buchstaben die gleiche Ziffer zuzuweisen. Ein Wort wird nun kodiert indem alle darin enthaltenen Buchstaben einzeln gemäß Tabelle 5.1 umgesetzt werden, wobei nur die

ersten drei Stellen übernommen werden. Eine vollständige Ausgabe des Algorithmus erzeugt also immer eine Zeichenkette der Form A-000, wobei A für den Anfangsbuchstaben steht und 0 für eine Ziffer von 0-7. Sollte ein Wort nicht genug Zeichen

| Ziffer | Eingangszeichen |
|---|---|
| 0 | A, E, H, I, O, U, W, Y |
| 1 | B, F, P, V |
| 2 | C, G, J, K, Q, S, X, Z |
| 3 | D, T |
| 4 | L |
| 5 | M, N |
| 6 | R |

Tabelle 5.1.: Soundex Kodierung

enthalten, dass drei Codes gefunden werden können, so werden die noch freien Stellen mit der Ziffer '0' aufgefüllt. Bezüglich der Kodierung anhand der Tabelle gibt es noch einige Sonderregeln zu beachten. Doppelbuchstaben werden wie ein einzelnes Vorkommen des Zeichens behandelt. Ebenso werden aufeinanderfolgende Buchstaben mit der gleichen Codeziffer nur als ein Zeichen kodiert, diese Regel wird auch befolgt wenn diese beiden Zeichen von einem H oder W getrennt werden. Sollten andere nicht zu kodierende Zeichen zwischen den Buchstaben sein, also Vokale beziehungsweise das 'Y', so wird das zweite Vorkommen des Zeichens ganz normal kodiert, es kann in der Ausgabe also durchaus die selbe Ziffer mehrmals hintereinander auftreten. In der deutschen Sprache gilt als Erweiterung dieser Regeln, dass Umlaute wie Vokale behandelt werden, und das 'ß' wird wie ein normales 'S' betrachtet wird.

### 5.1.2. Metaphone

Die nächste zu betrachtende Kodierung ist der Metaphone Algorithmus[Bla07], der erst 1990 beschrieben wurde. In dieser Kodierung werden alle Konsonanten eines Wortes auf 16 mögliche Zeichen abgebildet. Dieser Algorithmus kodiert, im Gegensatz zu Soundex, nicht Zeichenweise, sondern jeder Konsonant in dem Eingangswort wird betrachtet und anhand seiner Umgebung eine Fallunterscheidung getroffen auf welches Zeichen er abzubilden ist. Am Beispiel des Eingangszeichens 'D' betrachtet lautet die Kodierungsvorschrift:
J, wenn in -dge- -dgy- -dgi-
T, in allen anderen Fällen
Beim Metaphone ist die Ausgabe nicht in ihrer Länge einheitlich, richtet sich also je nach Anzahl der zu kodierenden Zeichen. Er arbeitet genauer als Soundex, und durch die Fallunterscheidungen werden speziell auszusprechende Zeichenkombinationen ge-

sondert behandelt. Da diese Definitionen auf die englische Aussprache hin optimiert wurden, sind die Ergebnisse vor allem in dieser Sprache präzise.

### 5.1.3. NYSIIS

Der NYSIIS Algorithmus [Bla09] (New York State Identification and Intelligence System), 1970 veröffentlicht, basiert auf empirischen Untersuchungen zur Aussprache von Namen. Zu Begin der Kodierung werden gesondert der Beginn und das Ende der Zeichenkette angepasst. Anschließend werden bestimmte Zeichenkombinationen ähnlich wie beim Metaphone ersetzt. Im Gegensatz zu den ersten beiden beschriebenen Algorithmen werden beim NYSIIS die Vokale mitkodiert, wobei alle Vokale immer als 'A' dargestellt werden. In seiner Arbeitsweise ist der NYSIIS dem Metaphone sehr ähnlich, die einzelnen Schritte und Fallunterscheidungen unterscheiden sich insofern, dass ein anderer Fokus bei der Definition der Prozesses gesetzt war. Auch hier gilt wieder, dass der Algorithmus auf die englische Sprache hin optimiert ist.

### 5.1.4. Kölner Phonetik

Für die deutsche Sprache entwickelt wurde das Kölner Verfahren[Pos69], auch die Kölner Phonetik genannt, welches 1969 veröffentlicht wurde. Ähnlich wie bei Soundex werden die Zeichen der Eingabe auf verschiedene Ziffern abgebildet, allerdings wird beim Kölner Verfahren auch der Anfangsbuchstabe mitkodiert. Die Kodierung erfolgt zum Teil zeichenweise, allerdings mit einigen Ausnahmen, um im deutschen mit eigener Aussprache versehene Buchstabenkombinationen zu erfassen, z.B. 'ph'.

Die Kodierung gemäß dem Kölner Verfahren erfolgt zunächst nach Tabelle 5.2 (angelehnt an [Wil05]), wobei eine Ausnahme existiert, wenn das zu kodierende Wort mit 'Cl' oder 'Cr' beginnt wird das führende 'C' durch eine 4 statt einer 8 kodiert. Die so erhaltene Ziffernfolge wird weiterhin bearbeitet, dass alle mehrfach hintereinander folgenden Auftreten einer Ziffer durch eine einzige Stelle ersetzt werden. Abschließend werden noch alle Ziffern 0 aus der Folge entfernt. Das so erhaltene Ergebnis repräsentiert das Eingangswort gemäß der Kölner Phonetik.

### 5.1.5. Alternativen

Eine andere Herangehensweise an das Problem Datensätze zu einzelnen Patienten auch dann noch finden zu können, wenn in einem der Stringattribute ein ähnlicher Wert eingetragen ist, der aber nicht identisch zu dem Wert im Vergleichsdatensatz ist, wäre die Distanzberechnung zwischen zwei Strings. Der Levenshtein Algorithmus [Lev66] nimmt als Eingabe zwei Zeichenketten entgegen und errechnet die Mindestanzahl an Editieroperationen einzelner Zeichen um eine Eingabe in die Andere

| Ziffer | Eingangszeichen | |
|---|---|---|
| 0 | A, E, I, J, O, U, Y | |
| 1 | B | |
| | P | nicht vor H |
| 2 | D, T | nicht vor C, S, Z |
| 3 | F, W, V | |
| | P | vor H |
| 4 | G, K, Q | |
| | C | vor A, H, K, O, Q, U, X außer nach S, Z |
| | X | nicht nach C, K, Q |
| 5 | L | |
| 6 | M, N | |
| 7 | R | |
| 8 | S, Z | |
| | C | nach S, Z |
| | | nicht vor A, H, K, O, Q, U, X |
| | D, T | vor C, S, Z |
| | X | nach C, K, Q |

Tabelle 5.2.: Kodierung gemäß dem Kölner Verfahren

umzuwandeln. Das Problem hierbei ist der, im Vergleich zu einer einfachen Hashwertberechnung wie dem Soundex, sehr hohe Ressourcenaufwand um die Bewertung der Ähnlichkeit der Eingabewerte vorzunehmen. Im Kontext eines verteilten MPI Systems kommt eine Anwendung dieser Technik allerdings nicht in Frage, da die Datensätze der Patienten nicht im Klartext verschickt werden, und auf den weiter bearbeiteten Daten keine Nähe der Ursprungszeichenketten mehr berechenbar ist.

# 6. Lösungsansatz

Aus der Untersuchung der bestehenden Systeme geht eine ganze Reihe von Parametern hervor, die es zu beachten gilt. Ebenso müssen einige Entscheidungen über die Art der Umsetzung verschiedener Funktionen getroffen werden. Zum Teil stehen hierfür verschiedene Algorithmen zur Verfügung mit Hilfe derer einzelne Prozesse vollzogen werden können.

Um später auf den identifizierenden Daten von Patienten Vergleichsoperationen durchführen zu können, muss gewährleistet sein, dass die Inhalte bestimmten Attributen zugeordnet sind. Da nicht alle bestehenden Systeme bereits Attributsnamen aus einem festzulegenden Standard, beispielsweise HL7, verwenden, muss an dieser Stelle ein Datenwrapper eingesetzt werden über den eine Standardisierung erfolgen kann. Am Beispiel des Feldes 'HL7:Date of Birth' bedeutet dies, dass im zugrunde liegenden Informationssystem das Feld gefunden werden muss in dem die Geburtsdaten der Patienten gespeichert sind. Diese Inhalte müssen im passenden Format in das 'Date of Birth' eingebracht werden, ein reiner Kopiervorgang ist in diesem Fall nicht ausreichend da gegebenenfalls eine Konvertierung auf YYYYMMDD erfolgen muss. Zum Vergleich dieser Daten muss genau definiert werden auf welchen Attributen dieser stattfinden soll, da diese alle für jeden Datensatz vorhanden sein müssen. Gleichzeitig sollten es Attribute sein die sich nicht ändern können, so ist zum Beispiel der Mädchenname besser für einen solchen Vergleich geeignet als der Nachname. Hierbei ist es aber wenig wahrscheinlich das in allen Datensätzen das Attribut Mädchenname gesetzt ist. Die Attributsauswahl für den Vergleichsprozess ist somit kein trivialer Vorgang.

Um nicht alle zum Vergleich herangezogenen identifizierenden Attribute eines Patienten komplett übertragen zu müssen, was aus Datenschutzgründen auch nicht möglich wäre, müssen die Daten zuerst verarbeitet werden bevor sie das lokale System verlassen können. Um sowohl den Kommunikationsaufwand gering zu halten, als auch das Datenvolumen in der möglichen Speicherung fertig verarbeiteter Werte minimieren, ist es sinnvoll mit Hilfe einer geeigneten Hashfunktion über alle für die Vergleichsoperation relevanten Daten einen Wert zu errechnen. So ist es bei Verwendung einer kryptographisch sicheren Hashfunktion nicht möglich aus den Hashwerten Rückschlüsse auf die Daten des Patienten zu gewinnen. Gleichzeitig kann das empfangende System, durch äquivalente Berechnung eines Hashwertes des dort vorhandenen Datensatzes, einen Vergleich der beiden berechneten Werte durchführen. Aufgrund der geringen Kollisionswahrscheinlichkeit solcher Hashfunktionen treten Übereinstim-

mungen der Hashwerte bei verschiedenen Eingangsdaten nur extrem selten auf. Es ist nicht ausreichend auf Basis identischer Hashwerte direkt zu schlussfolgern, dass es sich um den gleichen Patienten handelt. Allerdings ist es sinnvoll in diesem Moment weitere Überprüfungen einzuleiten, da es sich hier bereits um einen konkreten Anhaltspunkt handelt, können auch einige Ressourcen darauf verwendet werden die Gleichheit der Patienten zu bestätigen, ohne inakzeptable Performanceprobleme zu verursachen.

Durch den Einsatz solcher Hashwerte ergibt sich allerdings ein neues Problem bei der Durchführung der Vergleichsoperationen im Hinblick auf die Eingangsdaten. In der Sun Matching Engine existiert ein konfigurierbarer Wert der als 'Matching Threshold' deklariert ist. Durch diesen ist es möglich festzulegen zu welchem Grad zwei Datensätze übereinstimmen müssen um als potentieller Treffer behandelt zu werden und gegebenenfalls weitere Schritte einzuleiten. Die Intention nicht nur bei absolut 100%iger Übereinstimmung der Daten weitere Aktionen zu starten liegt unter anderem darin, dass es möglich wäre das beispielsweise ein Tippfehler in den Daten existiert. Durch die Bildung des Hashwertes sind minimale Abweichungen an den Eingangsdaten allerdings nicht mehr am Ergebnis ersichtlich, so kann sich ein MD5-Hashwert eines Strings bereits bei nur einem unterschiedlichen Zeichen in allen Stellen verändern. Um die Qualität der Suche von gleichen Patienten zu verbessern ist es also nötig, die Eingangsdaten für die Hashbildung nach Abschluss der Standardisierung weiter zu bearbeiten. Im Falle des Namens des Patienten kann dies zum Beispiel die phonetische Kodierung mit Hilfe eines passenden Algorithmus wie des Soundex oder Metaphon sein. Auf einer genau definierten Menge dieser verarbeiteten Daten wird nun die Hashfunktion ausgeführt. Auf den so erhaltenen Werten findet die Suche nach Äquivalenten statt. Falls diese anonymen Werte, mit Hilfe derer eine Suche nach identischen Personen möglich ist, auf einzelnen Attributen generiert werden, bezeichnet man diese auch als Kontrollnummern [DLAE07].

Eine grundsätzliche konzeptionelle Entscheidung ist die Wahl von Zeitpunkt und Ort des Vergleichsprozesses dieser Kontrollnummern. Die Möglichkeiten in diesem Fall sind zum einen, zu jedem Patienten der neu registriert wird, nach Aufnahme dessen identifizierender Daten, die Kontrollnummern zu erstellen und diese an eine noch zu bestimmende Menge von Systemen zu versenden. Dies wäre ein push-basierter Ansatz. Außerdem ist es sinnvoll die fertig erstellen Werte mit in dem lokalen Datensatz zu speichern. Wenn nun ein System eine Kontrollnummer von einem anderen Teilnehmer erhält, so kann im hier vorhandenen Datenbestand nach eben dieser Kontrollnummer gesucht werden. Falls kein Treffer vorliegt ist keine weitere Aktion erforderlich. Sollte die Kontrollnummer vorhanden sein, wird ein Prozess in Gang gesetzt für den es wiederum mehrere denkbare Realisierungsmöglichkeiten gibt auf die im laufe dieses Kaptitels noch näher eingegangen wird. In Abbildung 6.1 sieht man das eine Referenzliste zurückgeliefert wird, die Daten aus System-ID und Patienten-ID aller bekannten Vorkommen des Patienten mit dem gerade empfange-

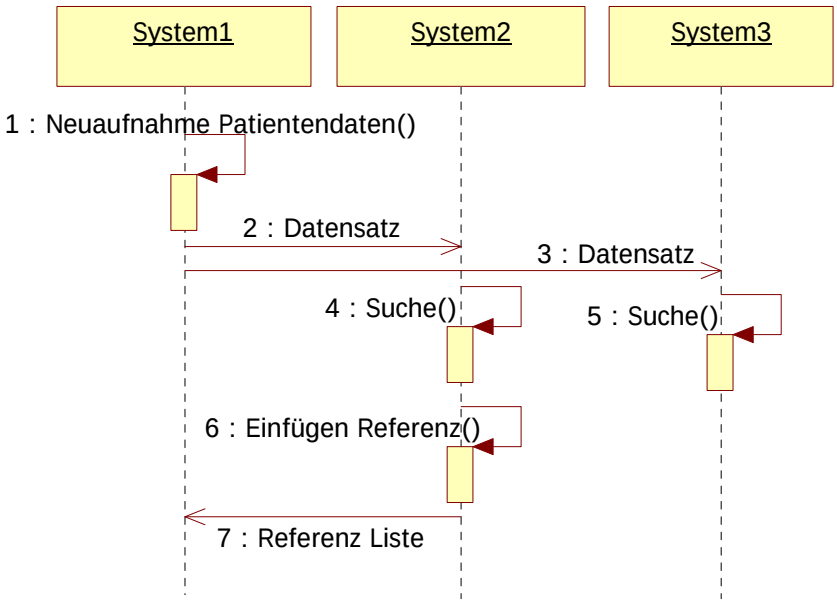


Abbildung 6.1.: Verteilte Referenzierung

nen Hashwert enthält. Dieses Vorgehen würde allerdings bedeuten, dass Patienten für die der gleiche Hashwert erstellt wird automatisch miteinander referenziert werden. Da keine 100%ige Kollisionsfreiheit besteht ist dieses Vorgehen nicht zulässig.

Der zweite mögliche Zeitpunkt zu dem der Vergleichsprozess gestartet werden kann ist der Aufruf einer Suche zu Daten eines Patienten. In diesem Fall ist es trotzdem sinnvoll den Hashwert zu einem Patienten bereits nach Aufnahme der Daten zu berechnen, um nicht im Falle einer eingehenden Suchanfrage eine große Menge an Daten verarbeiten zu müssen. Allerdings wird der Wert nur gespeichert und nicht versendet. Wenn nun Daten zu diesem Patienten benötigt werden, die nicht im System vorhanden sind, beispielsweise ein Vorbefund von einem anderen Arzt, so wird hierdurch der Vergleichsprozess gestartet der zur Referenzierung der Patienten-ID führen soll. In den bisher beschriebenen Fällen findet der Vergleichsprozess immer in den Fremdsystemen statt und nicht in dem initiierenden System, es ist aber auch möglich diesen Vorgang lokal zu vollziehen. Dazu müssen die Daten der anderen Systeme angefordert werden. Um nicht alle Daten empfangen zu müssen wird in diesem Ansatz eine (Bereichs-)Anfrage von Attributen ausgeführt, in etwa 'Sende alle vorhandenen Hashwerte von Patienten die 197x geboren sind zurück', um die Datenbasis für den Vergleichsprozess zur Verfügung zu stellen. Mit den empfangenen Hashwerten wird nun der Vergleich mit den lokal vorhandenen Datensätzen durchgeführt und bei Übereinstimmung der weitere Prozess gestartet.

In den bisher beschriebenen Szenarien werden immer Hashwerte zusammen mit der ID des Systems übertragen. Es ist also jedem System möglich außer den eigenen Datensätzen alle Datenpaare aus fremden SystemID's und Hashwerten dortiger Datensätze zu speichern, um im Falle eines Suchvorganges bereits eine größere Datenbasis zur Verfügung zu haben. Ob Daten auf Vorrat gehalten werden, kann als Teil der individuellen Konfiguration des einzelnen Systems festgelegt werden. So wird verhindert das relativ kleine Systeme von großen Datenmengen überlastet werden, gleichzeitig können Systeme mit genügend Ressourcen diese nutzen um ihre Vorgänge zu beschleunigen. Durch Senden derartiger Konfigurationsdaten an die benachbarten Teilnehmer kann sich ein großes KIS zum Beispiel als Indexspeicher bei kleineren Systemen bekannt machen. Diese können also bei Suchanfragen zuerst an denjenigen Knoten senden der einen Index führt. Ein solches Konzept innerhalb eines Peer-to-Peer-Netzes wird als Superpeer bezeichnet, wobei der Superpeer hier sowohl indexführende Einheit, als auch Makler für Suchanfragen darstellt. Wenn die Superpeers eine Adresstabelle anderer Superpeers führen, so kann eine Suchanfrage sehr leicht auf große Teile des Netzes ausgeweitet werden. Um die Flut an Anfragen einzudämmen können ähnliche Modelle wie bei DNS Systemen eingesetzt werden, so kann eine Anfrage zu einem Patienten beispielsweise zuerst an den Superpeer gerichtet werden der dessen Wohnort am nächsten liegt. Die Bindung eines teilnehmenden Systems an einen Superpeer kann hierbei statisch sein, es existieren aber auch Architekturen die dynamisch den optimalen Superpeer für ein System wählen und die Anfragen an diesen senden [GEvS07].

## 6.1. Standardisierung

Bevor Datensätze zu einzelnen Patienten in einem verteilten MPI System genutzt werden können, ist es zunächst erforderlich die darin enthaltenen Daten aufzubereiten um sie an die Anforderungen des Gesamtsystems anzupassen. Der Bestand an Patientendaten der in den verschiedenen Informationssystemen vorhanden ist, auf dem vom zu bildenden MPI System Verknüpfungen durchgeführt werden sollen, ist innerhalb jedes Teilsystems von der lokalen Konfiguration abhängig. Je nach Fachrichtung der medizinischen Einheit in der sich das Daten führende System befindet, unterscheiden sich die medizinischen Daten innerhalb der Datensätze stark. Dies ist jedoch für den MPI nicht von Belang, da dieser lediglich auf den Attributen arbeitet welche die den Patienten identifizierenden Daten beinhalten. Die medizinischen Inhalte werden nur als Nutzlast betrachtet. Bei den identifizierenden Daten sind folgende Punkte ebenfalls von der jeweiligen Implementierung abhängig: In welchem Format diese vorliegen, die Namensgebung der Attribute, welche dieser Attribute als erforderlich eingestuft sind und die Gesamtmenge der Attribute die gespeichert werden kann. Auch wenn es sich in allen Systemen um die Entität 'Patient' welche durch

diese Attribute charakterisiert wird handelt, unterscheiden sich die Datensätze doch potentiell so stark voneinander das eine Vergleichoperation direkt auf diesen Daten nicht möglich ist. Welche Daten zu einem Patienten als identifizierend angesehen werden ist eine Designentscheidung die sich ebenfalls zwischen den Implementierungen variieren kann. Im HL7 Standard werden 30 Attribute im PID Segment [Tabelle A.1] geführt wovon ein großer Teil als optional markiert ist. Lediglich die beiden Attribute 'HL7:Patient ID (Internal ID)' und 'HL7:Patient Name' sind als erforderlich eingestuft.

Zunächst muss also definiert werden auf welche Attribute es dem Vergleichsprozess möglich sein soll zurückzugreifen. Genau diese gilt es zu standardisieren, wohingegen alle anderen Attribute in ihrer ursprünglichen Form verbleiben können ohne die Funktionalität einzuschränken. Bei den in Kapitel 4 betrachteten Ansätzen haben die beiden Frameworks PIX und PIDS nicht genau festgelegt wie der Vergleichsprozess umgesetzt werden soll, und auch nicht auf welchen Attributen dieser vollzogen wird. Bei PIX werden dem CRM, in dem die Vergleichsprozesse ablaufen, als Datenbasis die Inhalte des IHE Patient Identity Feed zur Verfügung gestellt. In diesem sind zumindest die im PID Segment einer HL7 Nachricht erforderlichen Attribute enthalten. Alle weiteren erforderlichen Attribute müssen von der jeweiligen Implementierung definiert werden. Eine gültige Implementierung des CRM setzt allerdings voraus, dass zumindest alle Attribute aus Tabelle 4.2 gespeichert werden können, da im Falle einer Vergleichsoperation die interne ID eines Patienten nicht verwendet werden kann, und ein Vergleich lediglich auf Basis des Namens keine zufriedenstellenden Ergebnisse liefert. In der Spezifikation von PIDS werden die Attribute, welche übertragen werden sollen, bei der Konfiguration des Interfaces festgelegt. Falls das vordefinierte Modul OMG HL7Version2_3 verwendet wird, werden automatisch alle Attribute der HL7 PID Segment Felder 5-30 definiert. Die Auswahl der erforderlichen Attribute wird hierbei weitgehend der Anwendung überlassen, lediglich Name, Geburtsdatum und Geschlecht werden zwingend vorausgesetzt. Im Sun MPI finden umfassende Standardisierungen, auch von Attributen die nicht an die Sun Matching Engine für den Vergleichsprozess weiter gesendet werden, statt. Der Grund hierfür ist die Unterstützung verschiedener Suchtypen auf diesen Daten. Unter anderem sind Bereichsanfragen definiert die nur auf standardisierten Daten möglich sind. Im Falle des verteilten MPI werden die Daten der Patienten nicht in auslesbarem Format in den Katalogen abgelegt, daher werden derartige Suchanfragen nicht unterstützt und es ist nicht erforderlich Daten die nicht für den Vergleichsprozess herangezogen werden sollen zu standardisieren. Beim Sun MPI werden fünf Attribute an die Matching Engine übergeben auf Basis derer die Vergleichsoperationen durchgeführt werden. Hierbei sind Vor- und Nachname des Patienten, die Sozialversicherungsnummer, das Geburtsdatum und das Geschlecht als Standard definiert, wobei auch dieser MPI konfigurierbar ist und die Auswahl der Attribute variiert werden kann.

Überschneidend werden in allen betrachteten Ansätzen der Vor- und Nachname

des Patienten, sowie das Geburtsdatum und das Geschlecht als identifizierende Attribute angesehen. Die Sozialversicherungsnummer wird von PIX und der Sun MPI ebenfalls standardmäßig herangezogen. Da diese im HL7-PID Segment im Feld 19 vorgesehen ist, kann bei Verwendung des HL7 Moduls auch in PIDS dieses Attribut in den Vergleichsprozess aufgenommen werden. Bei PIX sind im Anforderungsprofil an den CRM noch weitere Attribute aufgeführt, allerdings sind dies zum Teil veränderliche Werte wie Telefonnummern, oder mit hoher Wahrscheinlichkeit in vielen Datenbeständen nicht vorhandene Werte wie die Führerscheinnummer des Patienten, und werden daher in diesem Kontext als nicht geeignet betrachtet.

Alle Attribute die für die Vergleichsoperation in Frage kommen müssen standardisiert werden. Des Weiteren ist es sinnvoll nur Attribute auszuwählen, die in einem großen Teil der Informationssysteme als erforderlich gekennzeichnet sind, da leere Attributsfelder den Vergleichsprozess je nach Wahl des Algorithmus beeinflussen können. Wenn nun von den vier eben genannten Attributen, bei denen sich die untersuchten Ansätze einig sind, ausgegangen wird, so wäre es für eine Implementierung des Datenwrappers nicht günstig nur diese Attribute zu bearbeiten. Hierdurch müsste im Falle einer Anpassung der Vergleichsoperation, durch die diese auf andere beziehungsweise mehr Attribute zugreift, die gesamte Datenstruktur ebenfalls angepasst werden. Es ist also sinnvoll, dass ähnlich wie in der Spezifikation von PIDS, eine Reihe von Attributen so behandelt wird, dass es durch einfache Änderungen möglich ist diese als Parameter mit einzubeziehen. Für einen ersten Suchvorgang kann hierbei eine kleine Menge an mit hoher Wahrscheinlichkeit ausgefüllter Attribute verwendet werden. Auf der Ergebnismenge dieser können die kompletten vorhandenen identifizierenden Daten in den Datensätzen heran gezogen werden, um präzise Ergebnisse hinsichtlich der Gleichheit der betroffenen Patienten liefern zu können.

Im Falle von Feldern wie Geburtsdatum kann ein genaues Format vorgegeben werden in das die vorhandenen Werte konvertiert werden müssen. Beim Geburtsdatum ist es möglich jeden gültigen Wert in das Format YYYYMMDD zu übertragen. Im Gegensatz dazu muss bei der Sozialversicherungsnummer unterschieden werden aus welchen Staat der Datensatz stammt, da sich das Format hier International unterscheidet. In Deutschland wird beispielsweise eine 12-stellige Zeichenfolge vergeben, wobei in dieser das Geburtsdatum enthalten ist. Sofern die Sozialversicherungsnummer für einen Datensatz vorhanden ist, müsste dieses also nicht in einem separaten Attribut gespeichert werden. In der Definition der benötigten Attribute werden nationale Sonderfälle dieser Art nicht beachtet um die Portabilität nicht einzuschränken. Dieses Feld und auch die den Namen enthaltenden Attribute müssen also als Zeichenkette interpretiert werden, da sonst eine Sonderbehandlung für jeden möglichen Typ von Sozialversicherungsnummer gefunden werden muss. Hierbei muss eine einheitliche Behandlung in allen Systemen erfolgen, beispielsweise die Umwandlung aller Zeichenketten in ASCII Großbuchstaben. Im HL7 PID Segment ist nur ein Attribut vorhanden in dem der Name des Patienten enthalten ist. Dies wird von PIDS und PIX

übernommen, lediglich der Sun MPI nimmt eine Zerlegung in einzelne Attribute für Vor- und Nachname vor. Ebenfalls ist eine gesonderte Behandlung für Sonderzeichen in den Namen nötig, um das Ergebnis nicht abhängig vom ursprünglich verwendeten Zeichensatz zu unterscheiden.

## 6.2. Phonetische Kodierung

Als nächster Schritt zur Verbesserung der Qualität der Eingangsdaten für den Vergleichsprozess werden Attribute, sofern das Format geeignet ist, phonetisch kodiert. Ein solcher Vorgang ist auf Namen von Personen und Orten sinnvoll nutzbar, nicht aber auf Zeichenketten wie der Sozialversicherungsnummer. In Kapitel 5 wurden einige Algorithmen zur phonetischen Kodierung vorgestellt. Von den vorgestellten phonetischen Algorithmen ist der NYSIIS der einzige der speziell auf den Klang von Eigennamen hin optimiert wurde. Da es sich hierbei aber um die englischsprachige Aussprache der zu kodierenden Namen handelt, ist auch dieser nicht optimal für eine Anwendung im Rahmen dieses Konzeptes geeignet. Das Kölner Verfahren, welches speziell auf die deutsche Sprache hin abgestimmt wurde, wird daher als Algorithmus ausgewählt durch den die phonetischen Kodierungen erfolgen sollen. Die zur Kodierung ausgewählten Attribute werden nach Abschluss ihrer Standardisierung an den Algorithmus übergeben und die erzeugten Codes werden an die nachfolgenden Prozessschritte weitergeleitet.

## 6.3. Hashing

Nachdem die einzelnen Felder des Datensatzes standarisiert wurden und je nach Feldtyp gegebenenfalls auch eine phonetische Kodierung durchgeführt wurde, wird aus den Inhalten mit Hilfe irreversibler kryptographischer Hashfunktionen ein Hashwert, auch 'Digest' genannt, berechnet. Das Ziel dieser Operation ist die Generierung von Codes die für den Vergleichsprozess verwendet werden können, ohne das es möglich ist direkt aus einer solchen Kodierung Informationen über den betroffenen Patienten auszulesen. Eine Hashfunktion gilt als irreversibel wenn zumindest gilt, dass es nicht möglich ist für einen gegebenen Hashwert den dazu gehörigen Eingabewert zu finden, und es zusätzlich nicht möglich ist aus dem Hashwert eine zweite Eingabe zu konstruieren die ein identisches Ergebnis erzeugt. Besitzt eine Hashfunktion zusätzlich noch die Eigenschaft der Kollisionsfreiheit, so wird diese als kryptographisch bezeichnet. Dies bedeutet das es nicht möglich ist gezielt zwei voneinander verschiedene Eingaben zu erzeugen die den gleichen Hashwert ergeben.

Dieser Absatz soll kurz zwei der gängigsten kryptographischen Hashfunktionen beschreiben. Die Erste ist der MD5 Algorithmus(Message-Digest Algorithm 5) [Riv92],

bei dem für eine beliebige Eingabe ein 128 bit langer Hash ausgegeben wird. Als erster Schritt wird die Eingangsnachricht durch bitweises Auffüllen auf eine festgelegte Länge gebracht und die Größe der Ursprungsnachricht angehängt. Die so entstandene Nachricht wird in 512 bit lange Blöcke geteilt auf denen der Hauptschritt des Algorithmus durchgeführt wird. Nach der Initialisierung mit fest definierten Startwerten werden in vier Runden die Startwerte mit dem ersten Block der zu konvertierenden Nachricht mit Hilfe bitweiser Operationen verknüpft. Das Ergebnis dieses Vorgangs wird nun an Stelle der ursprünglichen Startwerte als Eingabe genutzt und zusammen mit dem zweiten Block der Nachricht kodiert. Dieser Vorgang wird wiederholt bis der letzte Block der Nachricht verarbeitet wurde, das letzte entstandene Ergebnis ist hierbei der MD5 Hash der Ursprungsnachricht.

Ein anderer sehr verbreiteter kryptographischer Hashalgorithmus ist der SHA-1 (Secure Hash Algorithm 1) [ED01], der ebenfalls auf dem bereits zuvor erschienen MD4 basiert. Das Auffüllen der Nachricht, sowohl in der Länge die erzielt wird, als auch in den Füllbits die dazu verwendet werden, geschieht hier analog zum MD5. Anschließend wird auch beim SHA-1 die Länge der Ursprungsnachricht angehängt. Anders als beim zuvor beschriebenen MD5 arbeitet der SHA-1 mit fünf 32 bit Worten innerhalb seiner Runden, daher auch der längere Ergebniswert von 160 bit. Die Initialisierung erfolgt auch hier mit im Standard definierten Startwerten. Nachfolgend durchläuft jeder der 512 bit Blöcke vier Runden verschiedener vorgegebener Manipulationsoperationen, und der Inhalt der fünf 32 bit Worte nach Durchlauf des letzten Blockes der Nachricht stellt den Ergebniswert dar. Der SHA-1 unterscheidet sich hierbei von seinem Vorgänger nur um einen bitweisen Linksshift an einem der Eingangswerte, welcher den Algorithmus allerdings nachweislich sicherer gegenüber potentiellen Kollisionsattacken macht.

### 6.3.1. Hashing über einzelne Attribute

Bei der Anwendung solcher Hashingverfahren im Kontext des verteilten MPI ist grundlegend zwischen zwei Möglichkeiten zu wählen. Die Erste dieser Möglichkeiten besteht darin, einzelne Attribute als Eingabewert für die Hashalgorithmen zu verwenden, was dem Prinzip der Kontrollnummern [TAS94] entspricht. Dies ermöglicht aus den zur Verfügung gestellten Daten zu wählen anhand welcher Attribute des Patienten die Vergleichsoperation durchgeführt werden soll. Sofern ein SHA-1 verwendet wurde, entsprechen je 160 bit des Datensatzes einem Attribut. Durch die Standardisierung der Reihenfolge dieser Attribute für alle teilnehmenden Systeme können durch Auswahl der passenden Bitsequenzen die Vergleichsoperationen flexibel angepasst werden. Am Beispiel der vier Attribute die übereinstimmend in allen untersuchten Ansätzen zur Patientenidentifizierung verwendet werden, also Vorname, Nachname, Geburtsdatum und Geschlecht besteht der kodierte Teil eines Datensatzes aus:

- Bit 0-159: Hashwert des Nachnamens
- Bit 160-319: Hashwert des Vornamens
- Bit 320-479: Hashwert des Geburtsdatums
- Bit 480-481: Geschlecht

Beim Geschlecht macht es Aufgrund der extrem kleinen Menge der Eingabewerte keinen Sinn einen Hash zu verwenden da dieser durch simples Testen der 4 möglichen Werte die im HL7 spezifiziert sind (männlich, weiblich, sonstiges, unbekannt), also eine vereinfachte Version eines Bibliotheksangriffs, bereits reversibel wäre. Die Eingangsdaten der anderen drei Attribute sind bereits standardisiert worden und die beiden Namensfelder wurden zusätzlich einer Phonetisierung unterzogen. Durch einen Abgleich mit dem bekannten Ergebniswert des Hashings einer leeren Eingabe, können in einzelnen Datensätzen nicht vorhandene Attribute bei der Auswahl bereits markiert und berücksichtigt werden. Daher ist es bei einem derartigen Vorgehen sinnvoll, die Menge der gespeicherten Attribute umfassend zu wählen, da lediglich für jedes Attribut bei der Erzeugung des Datensatzes ein Hashwert berechnet werden muss und pro Datensatz 20 Byte Speicherplatz verbraucht werden. Dieses Vorgehen ermöglicht auch den Vergleich von Datensätzen aus unterschiedlichen Domänen in denen sich die Auswahl der benötigten Attribute stark unterscheidet, da es auf den anonymisierten Werten noch immer möglich ist eine Schnittmenge an vorhandenen Attributen zu finden um auf eben diesen einen Vergleich auszuführen. Dies geschieht indem alle Attribute aus dem Vergleich ausgeschlossen werden bei denen einer der Eingangsdatensätze die Codierung der leeren Eingabe aufweist. Ein weiterer Vorteil ist die Möglichkeit auf einer vorhandenen Datenbank, welche lediglich aus Datenpaaren besteht einfache Abfragen zu realisieren. Diese Daatenpaare setzen sich zusammen aus anonymisiertem Datensatz und der zugehörigen ID, welche wiederum aus Domänen-ID und dortiger lokaler Patienten-ID besteht. Nicht möglich ist eine Bereichsanfrage bezüglich der enthaltenen Informationen, lediglich genaue Anfragen, auch nur auf einem Teil der Attribute sind realisierbar.

### 6.3.2. Hashing über Attributsgruppen

Die Zweite Möglichkeit eine kryptographische Hashfunktion zum anonymisieren der Patientendaten zu verwenden, ist aus allen für den Vergleichsprozess ausgewählten Attributen einen einzelnen String zu generieren, welcher anschließend als Eingabewert für das Hashverfahren verwendet wird. Bei dieser Variante ist es nicht mehr möglich auf den anonymisierten Daten nachzuvollziehen ob einzelne Attribute vorhanden sind oder nicht. Als Attributsauswahl muss also bereits vorab die Schnittmenge der in allen teilnehmenden Systemen gesetzten Attribute evaluiert werden, da

lediglich diese verwendet werden dürfen. Falls eines der einbezogenen Attribute leer ist, ist der Datensatz nicht mehr mit anderen vergleichbar, da selbst wenn alle gesetzten Attribute überein stimmen, der Hashwert nicht mit einem auch das fehlende Attribut enthaltenden Wert in Verbindung gebracht werden kann.

Die Verwendung eines einzelnen Hashwertes stellt also eine deutlich höhere Anforderung an die Datenqualität der Ursprungsdaten und erfordert bereits vor der Implementierung eine endgültige Auswahl der für den Vergleichsprozess herangezogenen Attribute. Bei der Realisierung mit je nach Attribut getrennten Hashwerten müssen lediglich alle für den Vergleich in Frage kommenden Attribute aufgenommen werden. Die Auswahl der für einen bestimmten Vergleichprozess herangezogenen Attribute kann im laufenden Betrieb erfolgen. Da bei Verwendung nur eines Hashwerts lediglich ein exakter Abgleich in allen Attributen erfolgen kann ist es nicht möglich, Ähnlichkeiten, beispielsweise alle identifizierenden Attribute außer dem Nachnamen stimmen überein, zu einem vorhandenen Datensatz festzustellen. Diese Ähnlichkeiten können genutzt werden, um zusätzliche Überprüfungen einzuleiten ob eine Namensänderung durch Hochzeit oder Ähnliches in der Zwischenzeit erfolgt ist. Ein kritisches Problem bei der Verwendung von einzeln kodierten Attributen ist die Verwundbarkeit gegenüber Bibliotheksangriffen. In diesem Fall existieren für jeden der Hashwerte eine deutlich geringere Menge an möglichen Eingangsdaten, im Vergleich zu der Konkatenierung aller relevanten Attribute die einer einzigen Hashfunktion übergeben werden. Eine weitere Verschlüsselung der entstandenen Datensätze löst diese Anfälligkeit, erfordert allerdings eine vorhandene Public-Key-Infrastruktur. Sollte dies nicht realisierbar sein, besteht noch die Möglichkeit einer Mischform bei der Attribute gruppiert, und in eben diesen Gruppen an die Hashfunktion übergeben werden. Hierbei kann ein Attribut auch in mehreren Gruppierungen auftreten, somit kann das Problem, auf ein einzelnes fehlendes Attribut in den Ausgangsdaten einzugehen, gelöst werden ohne die Menge der Eingangsdaten weit zu reduzieren. An dem zuvor vorgestellten Beispiel würde folgende Kodierung entstehen:

- Bit 0-159: Hashwert aus [Nachname;Vorname;Geburtsdatum]
- Bit 160-319: Hashwert aus [Vorname;Geburtsdatum;Geschlecht]
- Bit 320-479: Hashwert aus [Nachname;Geburtsdatum;Geschlecht]
- Bit 480-639: Hashwert aus [Nachname;Vorname;Geschlecht]

Diese Lösung erfordert mehr Rechenoperationen bei der Erzeugung eines Datensatzes, und auch die Länge des einzelnen Datensatzes steigt bei Aufnahme zusätzlicher Attribute am stärksten an, wobei hier nicht alle Permutationen der Attribute kodiert werden müssen. Durch diesen Ressourcenaufwand ist es möglich die Vorteile des Ansatzes, welcher sich an den Kontrollnummern orientiert, auch ohne eine vorhandene

PKI zu einem großen Teil zu realisieren. Bei der Erzeugung der Eingabestrings für die Hashfunktion wird, sofern eines der enthaltenen Attribute leer ist, der komplette Eingangswert verworfen und an die Stelle des Hashwerts der zum leeren String gehörige Wert geschrieben. Auf diese Art kann bei der Vergleichsoperation unterschieden werden, ob die Eingaben sich in ihren Werten unterscheiden, oder lediglich aus unterschiedlichen Attributskombinationen bestehen. Bei der Vergleichsoperation können also alle Bereiche die in einem der Eingangswerte den Hashwert des leeren Strings enthalten verworfen werden, und der Vergleich auf Basis der verbliebenen Attributsgruppierungen durchgeführt werden. Es ist hierbei noch zu entscheiden, ob die verbliebenen Gruppierungen im Falle einer Übereinstimmung aussagekräftig genug sind, oder weitere Überprüfungen vorzunehmen sind. Stimmen alle als erforderlich definierte Attribute überein, kann eine automatische Verknüpfung der Datensätze erfolgen. Bei einer teilweisen Übereinstimmungen der Attribute muss unterschieden werden, ob die voneinander verschiedenen Werte in einem der Fälle den 'nicht vorhanden' Wert tragen oder verschiedene Informationen enthalten sind. Für beide Varianten sind Schwellwerte bezüglich der Attributsgruppen zu definieren, in denen entweder ebenfalls eine automatische Verknüpfung, wie bei kompletter Übereinstimmung, durchzuführen ist, oder eine Benachrichtigung zur Überprüfung durch einen autorisierten Mitarbeiter zu versenden ist. So ist es sinnvoll alle als erforderlich für eine automatische Verknüpfung eingestuften Attribute als Gruppierung zu führen, bei deren Übereinstimmung direkt der Verknüpfungsprozess eingeleitet wird. Allen anderen Gruppierungen können positive und negative Gewichtungen zugewiesen werden. Im Falle der Übereinstimmung wird der positive Wert gewertet, sofern in einem der Datensätze der Wert nicht gesetzt ist erfolgt keine Wertung und bei verschiedenen Inhalten in dieser Gruppierung wird der negative Wert verrechnet.

## 6.4. Kommunikationsverfahren

Durch die Kombination der Ergebnisse der Hashverfahren mit der ID des Patienten und der ID der Domäne, in welcher dieser Prozess stattgefunden hat, entsteht ein Datensatz mit Hilfe dessen Vergleichsoperationen möglich sind. Diese Datensätze können anhand der enthaltenen ID's verknüpft werden, so dass gezielte Anfragen bei der entsprechenden Domäne möglich werden. Um diese Vergleiche durchführen zu können müssen die Datensätze zwischen den Teilnehmern ausgetauscht werden, dazu gibt es verschiedene Möglichkeiten. Zum einen ist hierbei festzulegen wann die Kommunikation der Datensätze erfolgen soll, zum anderen muss zwischen sogenannten push- und pull-Verfahren unterschieden werden. Eine Möglichkeit ist es, Daten zu einem Patienten wie bisher in diesem Kapitel beschrieben zu verarbeiten und das Ergebnis lokal zu speichern. Sobald ein System einen Suchvorgang startet, sendet es seine Anfrage in das Netz und erhält alle Datensätze um darauf die Vergleichsopera-

tionen durchführen zu können. Das Problem bei dieser pull Implementierung ist, dass auf den anonymisierten Daten eine Vorauswahl nur möglich ist, wenn bei der Anfrage zumindest Teile des fertig kodierten Datensatzes mit übertragen werden. Ohne eine solche Einschränkung würde das Netz allerdings bereits von wenigen Anfragen sehr stark ausgelastet, da jeder anfragende Knoten, wie in Abbildung 6.2 dargestellt, die kompletten Datensätze aller anderen Knoten benötigt um seine Vergleichsoperationen durchführen zu können, wodurch ein enormer Kommunikationsaufwand entsteht. Dem Aufruf von 'findCandidates' kann hierbei lediglich ein fertig kodiertes Attribut übergeben werden, was allerdings einen Teil der Vergleichsoperation bereits auf dem Fremdknoten nötig macht. Daher wird diese Methode in diesem Ansatz ohne Parameter aufgerufen was zur Folge hat das in der 'CandidateList' alle Datensätze des jeweiligen Systems enthalten sind. Des Weiteren müssen entweder Routing-Protokolle, wie sie in Sensornetzen zum Einsatz kommen genutzt werden, um die Anforderung an alle relevanten Knoten zu verbreiten. Dies wäre nicht nötig, wenn dem die Anfrage stellenden Knoten sind alle anderen Knoten im Netz bekannt sind, was Aufgrund von möglichen Erweiterungen und Umstrukturierungen des Netzes keine realisierbare Umsetzung erlaubt.

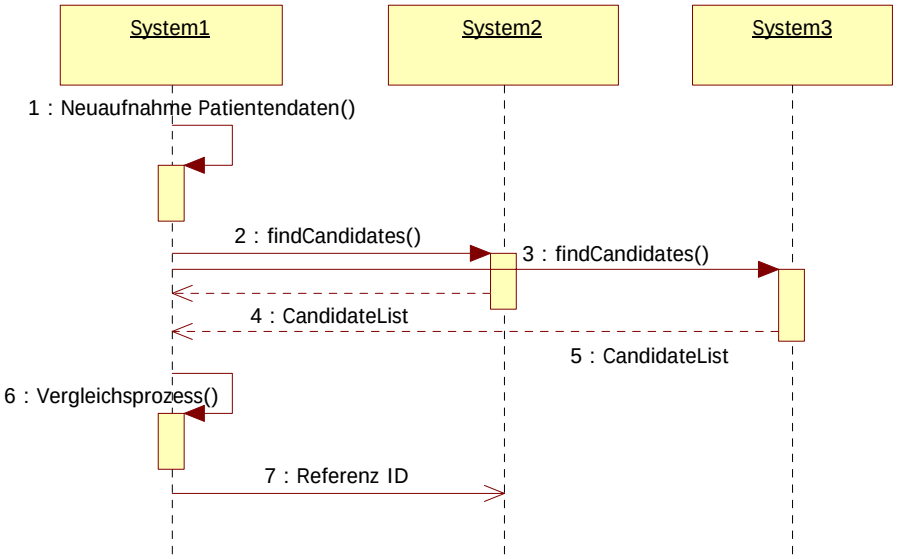


Abbildung 6.2.: Lokale Durchführung der Vergleichsoperationen

Durch die im Laufe dieses Kapitels beschriebenen Manipulationen an den identifizierenden Daten eines Patienten, welche erfolgen bevor diese in den Datensatz eingebracht werden, ist es nicht möglich anhand des Datensatzes die Identität des Patienten zu ermitteln, sondern lediglich eine Übereinstimmung mit anderen auf die gleiche Weise verarbeiteten Daten feststellbar. Weiterhin sind in den Datensätzen kei-

nerlei medizinische Informationen enthalten, sondern lediglich die identifizierenden Attribute über den betroffenen Patienten. Es ist also möglich diese Datensätze allen Teilnehmern an dem verteilten MPI zur Verfügung zu stellen ohne den Datenschutz der lokal vorhandenen Patientendaten zu verletzen. Die Verknüpfung der Patientendaten erfolgt also nicht auf dem System das Informationen zu einem Patienten abrufen soll, sondern dieses stellt den fertig errechneten Datensatz zur Verfügung und die Vergleichsoperationen werden an verschiedenen Stellen im Verbund ausgeführt. Des Weiteren muss unterschieden werden zu welchem Zeitpunkt diese Prozesse ablaufen sollen. Wenn als Zeitpunkt wie beim oben beschriebenen pull Ansatz derjenige Moment gewählt wird, in dem eine Suche zu einem Patienten gestartet wird, so kann keine Voraussage über die Antwortzeit dieser Suche getroffen werden, da diese von der aktuellen Auslastung der anderen Knoten im Netz abhängt. Daher ist es sinnvoll den Datensatz bereits zu versenden nachdem die Anonymisierung abgeschlossen ist, noch bevor eine konkrete Anfrage zu dem Patienten vorliegt. Die empfangenden Systeme können dann den eingegangenen Datensatz mit ihrem Datenbestand vergleichen, und im Falle einer Übereinstimmung die Verknüpfung vornehmen, sowie den sendenden Knoten über die zusätzliche Referenz informieren wie in Abbildung 6.3 dargestellt.

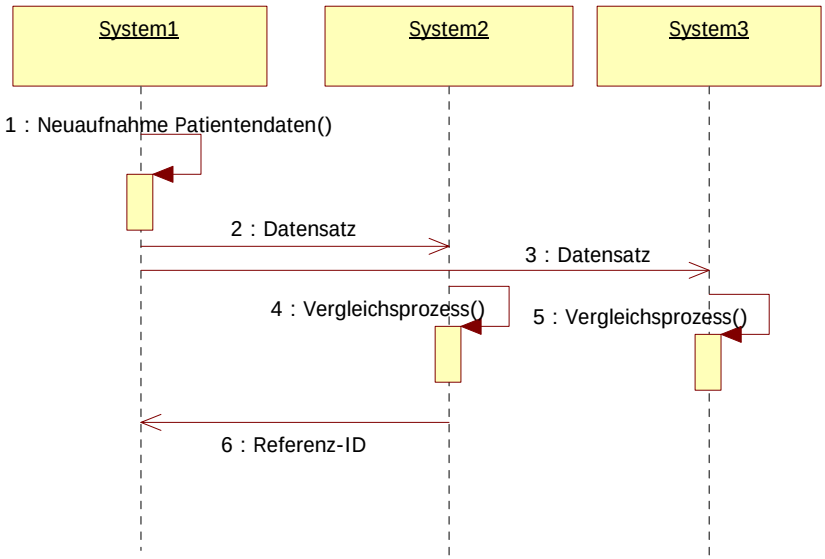


Abbildung 6.3.: Verteilte Durchführung der Vergleichsoperation

Das Problem der Überflutung des Netzes besteht allerdings noch immer, obwohl die Datenmenge pro aufgenommenem Patienten geringer ist, als beim zuvor dargestellten Ansatz. Dafür ist die Anzahl der Anfragen höher, da nun nicht erst der Prozess gestartet werden muss wenn tatsächlich eine Suchanfrage vorliegt, sondern für jeden in das System aufgenommenen Patienten. Die eigentliche Suche fällt bei

dieser Realisierung weg, da bereits die Referenz-ID bekannt ist, und nur noch der, im Rahmen dieser Arbeit nicht spezifizierte, Prozess zum Austausch der kompletten Patientendaten, also auch der medizinischen Informationen, zwischen den beiden Systemen in denen der Patient vorhanden ist erfolgen muss.

### 6.4.1. Inhomogene Netzstruktur

Bei klassischen MPI Systemen gibt es eine zentrale Stelle an die eine Suchanfrage gerichtet werden kann. Von dieser erhält man als Antwort die Referenz auf Patienten-ID's in anderen teilnehmenden Domänen, um dann Anfragen dorthin stellen zu können. Ein Problem bei der Entwicklung eines verteilten MPI ist, dass es keine zentrale Anlaufstelle gibt um eine Suche durchzuführen, da es sich lediglich um ein peer-to-peer Netzwerk handelt. Wenn keine zusätzlichen Mechanismen zur Unterstützung von Suchanfragen eingeführt werden, ist die einzige Möglichkeit die Anfrage an jeden Knoten im Netz zu senden. Da dies zu einer Überflutung des Netzes führen würde müssen also Vorkehrungen getroffen werden, um eine intelligente Suche zu ermöglichen. Wie voran beschrieben werden die Datensätze mit der ID eines Patienten und den kodierten identifizierenden Daten an eine Reihe von Systemen geschickt in denen anschließend die Vergleichsoperationen durchgeführt werden. Sofern jeder Datensatz an alle Knoten verschickt und dort direkt einem Abgleich unterzogen wird, so dass weitere Aktionen gestartet werden können, ist es jedem System nur möglich die eingehenden Daten mit eigenen Patientendaten zu vergleichen.

Um den Suchvorgang effizienter gestalten zu können muss zunächst die Möglichkeit geschaffen werden eingegangene Datensätze zu speichern, um diese für spätere Vergleichsoperationen heranziehen zu können. Hierbei ist es sinnvoll in leistungsstärkeren Systemen, bevorzugt also KIS, mehr Daten zu speichern als beispielsweise in einem Rechner einer kleinen Praxis eines niedergelassenen Arztes. Wenn ein Krankenhaus also sein am verteilten MPI teilnehmendes System so konfiguriert, dass es einen Katalog mit allen eingehenden Datensätzen generiert, so kann dieser Katalog auch von kleineren weniger leistungsstarken Systemen mitverwendet werden. Dieser Katalog kann als eine Art Repository für Hashwerte fungieren um den Suchprozess zu erleichtern (Abbildung 6.4). Das Prinzip solcher Makler die als normale Peers am Netzwerk teilnehmen, und zusätzlich Sonderaufgaben übernehmen von denen auch andere Peers profitieren, wird von Tanenbaum [AST08] als 'Superpeer-Konzept' bezeichnet. Hierbei ist zwischen einer statischen und einer dynamischen Bindung eines teilnehmenden Systems an einen Superpeer zu unterscheiden. Eine statische Bindung bedeutet im diesem Zusammenhang, dass ein teilnehmendes System das einen neuen Datensatz generiert hat, diesen an seinen zugehörigen Superpeer schickt. Dieser katalogisiert den Datensatz, führt die Vergleichsoperation auf seinem Datenbestand aus und sendet den Datensatz konfigurationsabhängig an andere Superpeers um ihn auch in deren Kataloge aufnehmen zu lassen. Sollte in der Vergleichsoperation eine

Übereinstimmung auftreten wird abhängig vom Grad der Übereinstimmung die Verknüpfung der Datensätze vorgenommen, oder eine Benachrichtigung zur Überprüfung generiert. In diesem Fall muss dem teilnehmenden System lediglich die Adresse des zugehörigen Superpeer bekannt sein, so dass alle Suchanfragen und Vergleichsoperationen über diesen abgewickelt werden können. Sofern dem Superpeer die Adressen aller anderen teilnehmenden Superpeers bekannt sind kann hierdurch der komplette Datenbestand des verteilten Systems erreicht werden. Bei einer dynamischen Anbindung der Systeme an die Superpeers muss dem System eine Liste aller in Frage kommenden Superpeers vorliegen. Für jede Operation wird einer dieser Superpeers als Kommunikationspartner ausgewählt. Hierzu stellt Garbacki [GEvS07] ein Verfahren vor, nach dem anhand vorangegangener Ergebnisse der bevorzugte Kommunikationspartner ausgewählt wird. Vereinfacht kann man sagen, dass hier derjenige Superpeer bevorzugt genutzt wird von dem mehr Suchanfragen erfolgreich beantwortet wurden. Ein Vorteil einer dynamischen Anbindung ist, dass bei Ausfall eines Superpeers nicht alle von diesem abhängenden Systeme vom Netz getrennt werden, die Knotenautonomie sowie Ausfallsicherheit und Verfügbarkeit werden also positiv beeinflusst. Bei einer statischen Anbindung existieren Vorteile in der einfacheren Authentifizierung der Systeme untereinander.

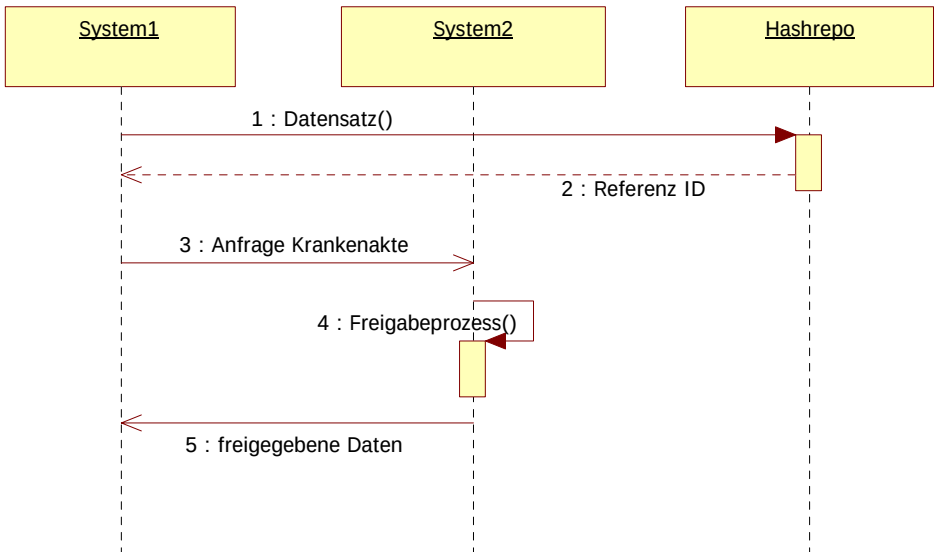


Abbildung 6.4.: Einsatz von Maklern in der Suche

Der Unterschied zu einer Replikation eines klassischen MPI Systems auf alle Superpeers ist hierbei, dass nicht alle Daten in jedem der Superpeers vorhanden sind. Für das anfragende System ist allerdings nicht ersichtlich woher die späteren Ergebnisse stammen, die Ortstransparenz ist also erfüllt. Je nach Größe des Netzes kann der

Datenbestand hierbei in unterschiedliche Teile zerlegt werden, wobei im Hinblick auf die Realisierung einer Suchfunktion im verteilten MPI unterschieden werden muss, ob man den für eine Speicherung des Datensatzes zuständigen Superpeer anhand bestimmter Attribute, oder auf Basis der Hashwerte auswählt. Am Beispiel der Hashwerte würde dies bedeuten, 'Die ersten 4 Byte des Datensatzes liegen zwischen a und b' wäre das Kriterium für einen bestimmten Superpeer einen Datensatz zu speichern. In allen anderen Fällen würde lediglich die Vergleichsoperation durchgeführt und danach der Datensatz verworfen sofern er von einem anderen Superpeer übertragen wurde, oder weitergeleitet falls er von einem angebundenen normalen Teilnehmer stammt. Um eine Partitionierung der Datensätze anhand der Inhalte bestimmter Attribute vornehmen zu können, müssen diese Attribute unkodiert auf den Superpeers vorliegen, da anders keine derartige Information mehr aus den Datensätzen zu gewinnen ist. Dies widerspricht den vorangegangenen Arbeitsschritten zur Anonymisierung der Patientendaten, die unter anderem aus Datenschutzgründen notwendig sind, und wird daher als nicht sinnvoll erachtet.

Als relevanter Parameter für die Partitionierung des Datenbestandes auf den Superpeers wird der Hashwert verwendet der die Gruppierung an Attributen repräsentiert bei deren Übereinstimmung die automatische Verknüpfung zweier Datensätze erfolgt. Die Information darüber, welcher Bereich in welchem Superpeer liegt, wird hierbei nur innerhalb der Superpeers ausgetauscht, was eine Änderung an dieser Partitionierung auf Grund von Wachstum des Gesamtsystems oder Reduzierung der Anzahl der Superpeers einfacher gestaltet. Die Partitionierung wird hierbei überschneidend gewählt, so dass ein Datensatz nicht nur auf einem Superpeer gespeichert wird um eine höhere Ausfallsicherheit und Verfügbarkeit zu erreichen.

Der Suchvorgang zu einem Datensatz der bei einem Superpeer eingeht startet also damit, dass der zugehörige Speicherort ausgelesen, und der Datensatz dorthin weitergeleitet wird. Außerdem startet der Superpeer den Vergleichsprozess zwischen dem eingegangenen Datensatz und seinem lokalen Datenkatalog. Ebenso läuft dieser Vergleich auf den Superpeers ab, die für die Speicherung des Datensatzes verantwortlich sind. Im Falle von Übereinstimmungen wird der Absender benachrichtigt. Sofern der Datensatz von einem anderen Superpeer gesendet wurde, wird dieser nach Katalogisierung der Daten auch benachrichtigt wenn keine Übereinstimmung gefunden wurde. Dieser sendet den Datensatz nun an die Systeme, in denen diejenigen Datensätze gespeichert sind deren erforderliche Attribute unvollständig sind. Gemäß der Definition in Kapitel 6.3.2. enthält der Bereich, in dem der für den Speicherungsort relevante Wert hinterlegt ist, den Hashwert des leeren Strings. Sollte auch in diesen Systemen keine Übereinstimmung festgestellt werden, kann der Datensatz noch zum Abgleich an alle anderen Superpeers weitergeleitet werden. Diese Anfrage wird mit niedrigerer Priorität behandelt, so dass ein System mit zu wenig Ressourcen um alle Anfragen abzuarbeiten zuerst diese Suchanfragen verwirft, da bei den anderen Stufen des Suchvorgangs die Wahrscheinlichkeit einer Übereinstimmung größer ist. Wenn

die Suchanfrage mit niederer Priorität gesendet wird, bedeutet dies unterschiedliche Werte in den Attributen, die für eine automatische Verknüpfung als hinreichend betrachtet werden. Nach Abschluss dieses Vorgangs liefert der Superpeer, an den der Datensatz ursprünglich gesendet wurde, eine Rückmeldung über die erfolgreiche Katalogisierung des Datensatzes, und falls Übereinstimmungen aufgetreten sind auch die zugehörigen Fremdschlüssel an das Ursprungssystem zurück.

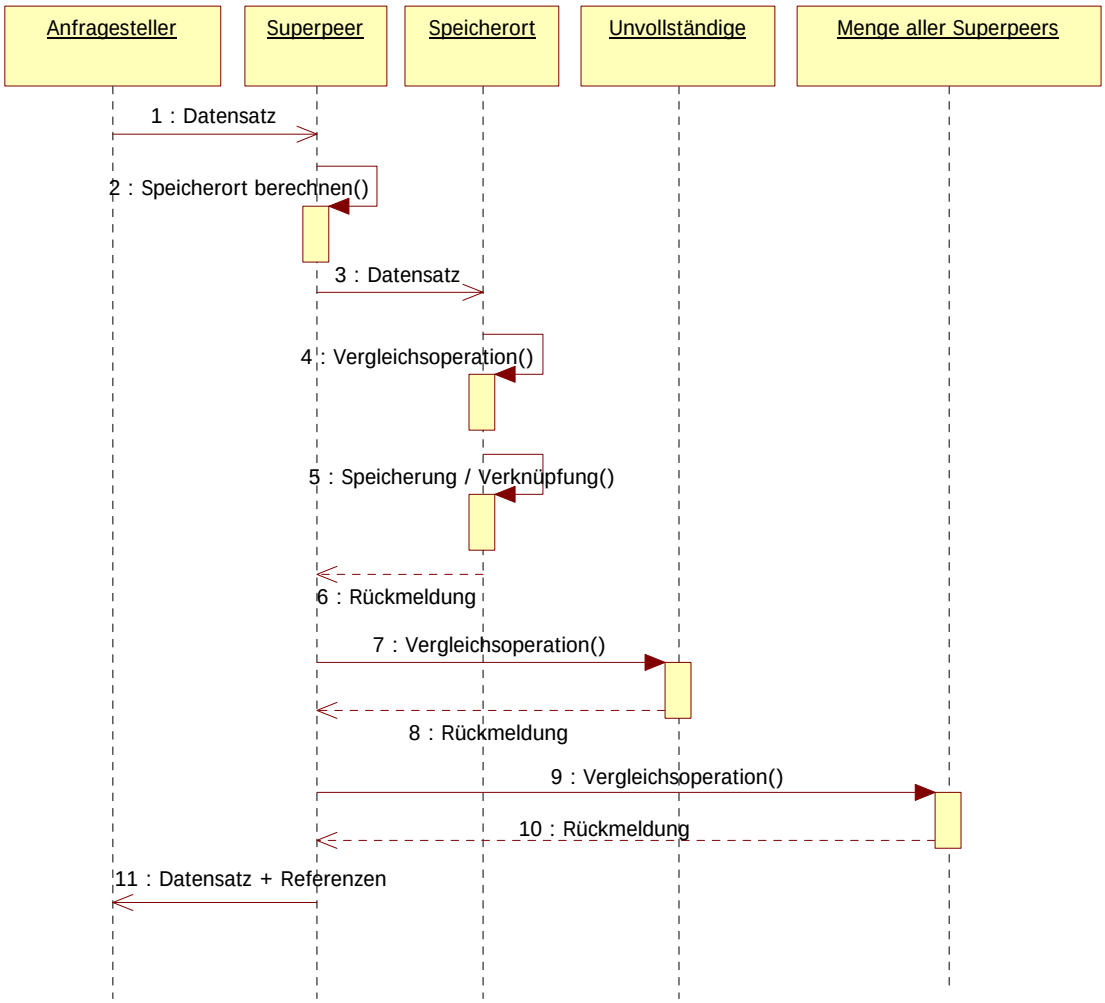


Abbildung 6.5.: Use-Case des Suchvorgangs

### 6.4.2. Use-Case

Der komplette Use-Case wird in Abbildung 6.5 dargestellt. In Schritt 5 wird unterschieden ob eine Übereinstimmung gefunden wurde. In diesem Fall wird der Datensatz mit dem vorhandenen verknüpft, anderenfalls liegt ein neuer Datensatz vor der lediglich gespeichert wird. Abhängig von der Rückmeldung die der initiierende Superpeer erhalten hat, ist der Suchvorgang, falls bereits eine Übereinstimmung gefunden wurde, nach Schritt 6 beendet. In diesem Fall werden noch die Speicherknoten der als Referenz gefundenen ID über den neuen Datensatz benachrichtigt. Da der Suchvorgang mit jedem Datensatz entweder über den kompletten Datenbestand oder bis zu einer vorhandenen Übereinstimmung durchgeführt wird, kann davon ausgegangen werden das die Liste der referenzierten ID's vollständig ist. Daher können alle Kommunikationsschritte ab Schritt 7 verworfen werden ohne das Ergebnis zu beeinträchtigen. Im Diagramm mit 'Unvollständige' gekennzeichnet sind diejenigen Speicherorte, an denen die Datensätze hinterlegt werden in deren, für den Vergleich primär herangezogenen Hashwerte, der Wert der leeren Eingabe hinterlegt ist. Der Unterschied bei der Kommunikation des Datensatzes zu diesen Knoten, im Gegensatz zu der vorangegangenen Kommunikation mit dem errechneten Speicherort, ist, dass direkt die Vergleichsoperation über eine Schnittstelle, mit dem Datensatz als Parameter, aufgerufen werden kann. Dies ist möglich da keine weiteren Aktionen, beispielsweise Speicherung, mit dem Datensatz durchzuführen sind. Sofern dieser Aufruf eine Übereinstimmung ergibt wird die übergebene ID in dem betroffenen Datensatz referenziert und die ID des betroffenen Datensatzes zurück geliefert. Andernfalls ist der Rückgabewert der ursprünglich übergebene Datensatz. Wenn der anfragende Superpeer lediglich den übergebenen Datensatz zurück erhält, so wird eine exakt gleiche Anfrage zu Schritt 7 wiederholt. Allerdings im nun folgenden Schritt an alle, bisher nicht am Suchvorgang beteiligten, bekannten Superpeers. Nach Erhalt aller Rückmeldungen oder dem Ablauf eines Timeouts, welcher nötig ist da diese niederprioren Anfragen auch verworfen worden sein können, liefert der Superpeer an den Anfragesteller ein Ergebnis zurück in dem der Ursprungsdatensatz, sowie alle gefundenen Referenz-ID's enthalten sind. Dadurch ist es dem System, welches die Suchanfrage gestartet hat, möglich, anhand der übergebenen Referenzen direkten Kontakt mit denjenigen Systemen aufzunehmen in denen Daten zum potentiell gleichen Patienten vorliegen.

## 6.5. Zusammenfassung

In diesem Kapitel wurde der Ablauf des Prozesses beschrieben, mit dem die identifizierenden Attribute eines Patienten bearbeitet werden. Ziel hierbei ist es die Möglichkeit zu schaffen, mit Hilfe von anonymen Daten, aus denen keinerlei Information über

die betroffene Person hervorgeht, die Übereinstimmung hinsichtlich der von einem Datensatz bezeichneten Person festzustellen. Zu diesem Zweck muss eine Auswahl getroffen werden, anhand welcher Attribute ein solcher Vergleich,unter Beachtung der Verfügbarkeit der einzelnen Daten, hinreichend zuverlässige Ergebnisse liefern kann. Sofern ein eine Person eindeutig identifizierendes Attribut, zum Beispiel eine Sozialversicherungsnummer, verfügbar ist, wird der Vergleichsprozess extrem vereinfacht. Allerdings darf nicht davon ausgegangen werden das ein solches Attribut in allen teilnehmenden Informationssystemen zur Verfügung steht. Die ausgewählten Informationen müssen je nach Datentyp unterschiedlichen Operationen unterzogen werden, um die stark differierenden Strukturen der gespeicherten Daten in den unterschiedlichen medizinischen Informationssystemen auszugleichen, um dadurch einen Vergleich dieser Daten zu ermöglichen. Hierzu zählt zum einen die Standardisierung des Formats in dem ein Wert gespeichert wurde, sowie abhängig von den Eingangsdaten eine Änderung an der Zuteilung der Informationen in die Attribute. Der Name eines Patienten ist in manchen Systemen kommasepariert in einem einzelnen Attribut gespeichert, in anderen wiederum stehen einzelne Attributsfelder für Vor- und Nachname zur Verfügung.

Durch die Anonymisierung der genutzten Identifikationsmerkmale mit Hilfe von kryptographischen Hashfunktionen wird die Möglichkeit geschaffen, Fremdsystemen Daten zur Verfügung zu stellen mittels derer festgestellt werden kann ob ein Patient mit identischen Merkmalen vorhanden ist, ohne dessen Identität zu veröffentlichen. Sofern ein identischer Datensatz vorliegt ist, sind dem System also automatisch die Daten des Patienten bekannt. Wird keine Übereinstimmung gefunden lassen sich keinerlei identifizierende Merkmale des Patienten auslesen.

Ein durch diesen Prozess neu entstandenes Problem, dass eine Ähnlichkeitssuche, welche direkt auf den identifizierenden Attributen ohne größeren Aufwand zu realisieren ist, nicht mehr umgesetzt werden kann. Es ist nicht mehr möglich eine Bereichsabfrage bei Daten oder die Festsetzung einer maximalen Distanz zwischen zwei Strings, zum Beispiel nach Levenshtein [Lev66], durchzuführen. In klassischen zentralisierten MPI Systemen können durch solche Vorgänge kleinere Eingabefehler, beispielsweise bei falsch verstandenen Namen oder Tippfehler in der Eingabe der Attribute, ausgeglichen werden. Da ein wichtiges Merkmal der verwendeten Hashfunktionen darin besteht, dass eine Ähnlichkeit der Eingabe sich nicht auf die ausgegebene Kodierung auswirkt, müssen neue Methoden verwendet werden um diese Korrekturmöglichkeiten eines MPI auch auf verteilte Systeme übertragen zu können. Eine solche Methode ist die phonetische Kodierung der Namensfelder in den identifizierenden Attributen des Patienten bevor diese an die Hashfunktionen übergeben werden. Dadurch soll die Korrektur kleinerer Eingabefehler bereits vor Übergabe der Daten an die Hashfunktion erfolgen, was durch den daraus ergebenen identischen Hashwert auch im Nachhinein feststellbar ist.

Unabhängig von der Verarbeitung der Identifikationsmerkmale besteht der zweite

Teil dieses Konzeptes aus einem Entwurf der regeln soll auf welche Art die entstandenen Datenpaare aus Hashwerten und zugehöriger ID Nummer ausgetauscht werden können, um den Vergleichsprozess in einem verteilten Umfeld nutzbar zu machen. Ohne die Nutzung von Katalogen ist in einem peer-to-peer Netz eine Suche nur durch Fluten des gesamten Netzes mit der Anfrage möglich. Um eine performantere Lösung umsetzen zu können wurde das Prinzip der 'Super-Peers' [AST08] genutzt. Auf diesen leistungsstarken Knoten wird ein verteilter Katalog von Datenpaaren verwaltet, mit Hilfe dessen die Identität eines zu kontaktierenden Knoten für eine einzelne Suchanfrage ermittelt werden kann, wobei deutlich weniger Kommunikationsaufwand benötigt wird.

Durch die Trennung des Zeitpunktes der Katalogisierung der Datensätze vom Suchvorgang, soll eine kürzere Antwortzeit der Suche erreicht werden. Um dies umzusetzen ist es nötig die in einem Knoten produzierten Datenpaare mit Hilfe von Push-Verfahren im Gesamtsystem an fest definierten Stellen zu hinterlegen auf die später im Rahmen der Suche zugegriffen werden kann. Die gleichen Kommunikationsabläufe die beim Verteilen der Daten genutzt werden, werden auch für den Vergleichsprozess verwendet. Dieser läuft in mehreren Stufen in den Knoten ab, die für das Bereitstellen der Kataloge verantwortlich sind.

# 7. Zusammenfassung

In diesem Kapitel soll noch einmal ein Überblick über diese Arbeit gewährt sowie die erlangten Ergebnisse zusammengefasst werden. Ein MPI soll den teilnehmenden Systemen die Möglichkeit gewähren an Informationen zu einem Patienten zu gelangen, die außerhalb ihrer eigenen Domäne vorhanden sind. Zu diesem Zweck ist es erforderlich, herauszufinden in welchen weiteren Domänen überhaupt Informationen zum betroffenen Patienten vorliegen. Außerdem ist es für eine gezielte Anfrage in einem Fremdsystem von großem Vorteil, wenn der dortige Primärschlüssel des Patienten, zu dem Daten abgefragt werden sollen, vorhanden ist. Diese beiden Aufgaben im Zuge der Informationsbeschaffung werden hierbei von einem MPI System übernommen. Der eigentliche Austausch der medizinischen Inhalte, der darauf folgend geschehen kann liegt hierbei nicht mehr im Rahmen der Spezifikation des MPI. Um ein Konzept für einen verteilten MPI erstellen zu können, müssen zuerst der Funktionsumfang und die Werkzeuge eines zentralisierten MPI Systems untersucht werden.

Zu diesem Zweck wurden unter anderem zwei Frameworks, die einen Rahmen zur Implementierung von MPI Systemen spezifizieren sollen, untersucht. Zum einen das IHE'Patient Identifier Cross-reference Integration Profile' (PIX), dessen Fokus vor allem darin liegt möglichst geringe Anforderungen an die teilnehmenden Systeme zu stellen. Dies soll erreicht werden indem der Großteil der zum Betrieb eines MPI nötigen Komponenten in einem zentralen Element, dem IHE'Cross-reference Manager' (CRM), umgesetzt werden. Die einzelnen Domänen stellen in diesem Profil die identifizierenden Merkmale ihrer Patienten über genau spezifizierte Transaktionen dem CRM zur Verfügung, der sämtliche dieser identifizierenden Informationen aus allen teilnehmenden Systemen speichert. Dadurch ist es möglich alle Vergleichsoperationen zentral im CRM zu bewerkstelligen. Dieser reagiert auf gefundene Übereinstimmungen indem er die verschiedenen Datensätze miteinander verknüpft. Durch diese im CRM geführten Verknüpfungslisten kann auf eine Anfrage eines Systems bezüglich eines Patienten reagiert werden, indem die für diesen Patienten vorhandenen Verknüpfungen zurück geliefert werden. Alternativ ist es auch möglich alle Systeme zu deren Patienten eine Übereinstimmung festgestellt wurde von den neuen Querverweisen in Kenntnis zu setzen. Dies ermöglicht den Systemen die weitere Kommunikation direkt zu führen, also immer diejenigen Systeme anzusprechen, welche Daten zum betroffenen Patienten enthalten und hierbei den Fremdsystemen auch die dortige Patienten-ID mitzuliefern. Um diese Funktionen gewährleisten zu können, wird im

Rahmen der Spezifikation ein Regelsatz definiert, der von allen teilnehmenden Systemen erfüllt werden muss. Hierbei wird darauf Wert gelegt, die Einschränkungen möglichst gering zu halten. Die genaue Auswahl der Algorithmen sowie der identifizierenden Attribute, welche für den Vergleichsprozess genutzt werden wird jedoch nicht getroffen, diese Entscheidung verbleibt also in der Hand der jeweiligen Implementierung. Ein MPI kann mit Hilfe von PIX umgesetzt werden, indem die ID einer festgelegten Domäne als Master-ID für alle Patienten fungiert. Hierbei ist es auch möglich eine künstliche Domäne zu erzeugen, die keinerlei eigene Daten verwaltet und lediglich zur Erzeugung von Surrogatschlüsseln genutzt wird, die in der ganzen IHE Cross-referencing Domain einzigartig sind.

Das zweite untersuchte Framework OMG'Person Identification Service' (PIDS) basiert, im Gegensatz zu PIX, nicht darauf eine funktionale Einheit zu definieren mit deren Hilfe die Prozesse realisiert werden, sondern spezifiziert Schnittstellen über die alle nötigen Transaktionen ausgeführt werden können. Zur möglichen Umsetzung eines MPI stellt das PIDS Profil zwei Schnittstellen zur Verfügung. Eine der beiden Schnittstellen ermöglicht das Hochladen eines Datensatzes, welcher die identifizierenden Attribute eines Patienten beinhaltet, in einen Katalog. Auf diesem wird zeitverzögert der Vergleichsprozess durchgeführt, worauf der hochladende Knoten keinerlei Einfluss mehr nehmen kann. Über die zweite Schnittstelle ist es möglich durch Übergabe eines identifizierenden Datensatzes die ID's aller zu diesem Patienten existierenden Querverweise zu erhalten. Allerdings wird auch hier nicht näher spezifiziert auf welchen Attributen oder mit Hilfe welcher Algorithmen die Vergleichsoperationen durchgeführt werden. Lediglich eine Mindestmenge an zur Verfügung stehenden Attributen wird gefordert, um die Wahl der genauen Vorgänge in der Implementierung nicht zu stark einzuschränken. In PIDS existieren gesonderte Schnittstellen, die bei der Umsetzung auch komplexer Rechtekonzepte hilfreich sind. So kann zwischen Zugriffsrechten auf Nutzerebene, die von der Authentifizierung der die Anfrage stellenden Instanz abhängen und Zugriffsrechten auf Datensatzebene unterschieden werden. So ist es möglich auf einzelnen Datensätzen voneinander unabhängig unterschiedliche Zugriffsrechte zu gewähren.

Beispielhaft für eine Implementierung eines Master Patient Index wurde der Sun MPI untersucht. Hierbei handelt es sich um ein Repository im Rahmen der Java CAPS, dessen Komponenten mit Hilfe von NetBeans genutzt werden können um einen MPI zu erzeugen und zu konfigurieren. Des Weiteren lässt sich die Kommunikationsanbindung des MPI an den Rest des Netzes relativ flexibel gestalten, wodurch es unter anderem möglich ist eine Umsetzung konform dem PIX Integrationsprofil zu realisieren. Die Vergleichsprozesse im Sun MPI werden durch die Sun Matching Engine durchgeführt, deren genutzte Attribute auch konfigurierbar sind, jedoch ist hier im Rahmen der Dokumentation des Repositorys eine Standardauswahl angegeben. Diese Daten überschneiden sich mit den Mindestanforderungen der beiden untersuchten Frameworks, was die Wichtigkeit dieser Attribute (Name, Vorname, Geburtsdatum

und Geschlecht) im Hinblick auf die Identifizierung eines Patienten unterstreicht.

Um die aus den untersuchten Ansätzen gewonnenen Einblicke in ein verteiltes Umfeld übertragen zu können, werden die zusätzlichen Anforderungen an einen verteilten MPI definiert. Durch den Wegfall eines zentralen Dienstes muss eine Lösung gefunden werden, mit deren Hilfe Datensätze in anderen Systemen lokalisiert werden können. Außerdem existiert kein Knoten mehr, dem alle anderen Systeme vertrauen, also im Rahmen eines MPI all ihre identifizierenden Daten über die vorhandenen Patienten zur Verfügung stellen. Mit Hilfe dieser Attribute werden in der Umsetzung eines MPI Systems die Vergleichsoperationen durchgeführt, es muss also ein Weg gefunden werden den Vergleich zu ermöglichen, ohne Informationen über die Identität des Patienten zu veröffentlichen.

Um dieses Problem zu lösen werden die Daten bevor sie das lokale System verlassen anonymisiert. Aus diesen Daten darf keine Information über den Patienten mehr gewonnen werden können, es muss allerdings möglich sein einen Vergleichsprozess auf ihnen durchzuführen. Um diese Eigenschaft zu gewährleisten werden die identifizierenden Attribute an eine kryptographische Hashfunktion übergeben. Die so gewonnenen Kodierungen erfüllen die beiden geforderten Eigenschaften. Es können keine Informationen über den Patienten mehr ausgelesen, bei gleichen Eingangsdaten die Übereinstimmung jedoch zweifelsfrei nachgewiesen werden. Bevor die Daten zur Kodierung an die Hashfunktionen übergeben werden ist es jedoch nötig eine Reihe von Manipulationen durchzuführen, da sich die Datentypen in der Speicherung je nach System unterscheiden können. Ebenso sind die Formate nicht in allen medizinischen Informationssystemen einheitlich, wodurch zunächst eine Standardisierung der Attribute notwendig wird. Im Zuge der Standardisierung werden die identifizierenden Merkmale des Patienten in einheitlich benannte Attribute gespeichert. Hierbei werden gegebenenfalls auch Inhalte einzelner Attribute aufgespaltet und in verschiedene Felder gespeichert, beispielsweise können in einem System kommasepariert innerhalb eines Feldes gespeicherte Namen in getrennte Felder für Vor- und Nachnamen unterteilt werden. Durch die Nutzung der Hashfunktionen, die speziell darauf ausgelegt sind bereits bei leicht verschiedenen Eingaben vollkommen unterschiedliche Kodierungen zu liefern, verliert der Vergleichsprozess die Option eine Ähnlichkeit der Eingangsdaten zu erkennen. Es besteht somit nicht mehr die Möglichkeit geringe Variationen, wie Tippfehler oder eine unvollständige Angabe des Geburtsdatums, in den Daten auszugleichen. Um diesem Nachteil entgegen zu wirken werden die Inhalte der zu kodierenden Attribute bevor sie an die Hashfunktion übergeben werden, bearbeitet. Hierbei werden zum Ausgleich von Tippfehlern, beziehungsweise von dem die Patientendaten aufnehmenden Personal falsch verstandene Informationen, phonetische Algorithmen eingesetzt. In diesem Konzept wurde Aufgrund ihrer besonderen Eignung für die deutsche Sprache die Kölner Phonetik als Methode zur phonetischen Kodierung gewählt. Bevor ein Attribut letztendlich anonymisiert wird, wurde der Inhalt standardisiert und phonetisch kodiert. Die so erhaltenen phonetischen Codes

können somit bei geringen Abweichungen übereinstimmen, wodurch ein Vergleich der durch die Hashfunktion kodierten Daten trotz unregelmäßiger Ursprungsdaten ermöglicht wird.

Für diesen gesamten Ablauf muss definiert werden, auf welchen Identifikationsmerkmalen diese Vorgänge durchgeführt werden müssen. Bei einer nachträglichen Anpassung des Algorithmus, mit dessen Hilfe die Vergleiche stattfinden, können nur in der Kodierung enthaltene Daten verwendet werden. Andernfalls muss der gesamte Bestand an fertig berechneten Datensätzen verworfen und neu berechnet werden. Als Eingangsdaten für den Vergleichsprozess werden hierbei Hashwerte verwendet die über jeweils mehrere Attribute gebildet wurden. Der 'primäre' Vergleichswert ist hierbei derjenige Hashwert, der über Vor-, Nachname und Geburtsdatum gebildet wurde, da diese Attribute die höchste Wahrscheinlichkeit besitzen in den einzelnen Patientendatensätzen erfasst worden zu sein. Sollte dieser bei verschiedenen Datensätzen übereinstimmen, kann davon ausgegangen werden, dass es sich um den gleichen Patienten handelt. Anderenfalls können weitere Werte überprüft werden, da beispielsweise durch eine Hochzeit ein Nachname geändert werden kann. Es werden daher mehrere Hashwerte über verschiedene Kombinationen unveränderlicher Attribute, wie Geburtsdatum, Geburtsort, Mädchenname der Mutter oder Sozialversicherungsnummer verglichen. Kann keine eindeutige Aussage darüber getroffen werden ob es sich um den gleichen Patienten handelt, können die Datensätze zur Bewertung an einen autorisierten Benutzer zur Bestätigung gesendet werden.

Um die Vergleichsprozesse durchführen zu können, muss die Kommunikation bezüglich der Datensätze geregelt werden. Hierbei werden Datenpaare bestehend aus ID des Datensatzes und den generierten Hashwerten versendet. Die ID eines solchen Datenpaares ist hierbei zweigeteilt und besteht aus der ID des Systems, aus dem der Datensatz stammt, und der lokalen ID des betroffenen Patienten in eben diesem System. Um zu verhindern, dass im Falle einer Suche nach einem identischen Patienten alle anderen Knoten befragt werden müssen, werden sogenannte Super-Peers eingeführt, die Kataloge mit Datenpaaren führen, und deren Gesamtheit die Funktionalität eines verteilten Datenbanksystems zur Verfügung stellt. Hierbei werden die Prozesse zur Katalogisierung und Verknüpfung der Datensätze, beziehungsweise zur Suche nach Patienten, voneinander losgelöst. Dadurch wird es beim Suchvorgang ermöglicht, direkt die Verknüpfungen zum betroffenen Patienten zu erhalten. Mittels der hierbei erhaltenen ID's kann direkt Kontakt zu den Systemen aufgenommen werden, in denen Datensätze zu diesem Patienten vorliegen. Der Verknüpfungsprozess wird initialisiert indem ein System ein Datenpaar an einen beliebigen Superpeer übergibt. Diese Knoten können anhand des primären Vergleichswertes des Datensatzes die potentiellen Speicherknoten für dessen Katalogeintrag berechnen. Daraufhin wird der Datensatz sowohl an die potentiellen Speicherknoten versandt, als auch der Vergleichsprozess auf dem lokalen Datenbestand gestartet. Sofern ein Superpeer einen Datensatz empfängt, dessen Speicherknoten er selbst ist, wird dieser Datensatz mit

dem lokalen Datenbestand abgeglichen. Zusätzlich wird unabhängig vom Ergebnis dieser Operation der erhaltene Datensatz in den Katalog aufgenommen und persistiert. Da dieser Prozess nicht an eine direkte Anfrage gekoppelt ist, sondern gestartet wird wenn ein teilnehmendes System die Daten eines neuen Patienten aufnimmt, ist es möglich diese Vorgänge auf den Superpeers nach der aktuellen Auslastung dieser auszurichten.

Durch die Nutzung dieser Prozesse ist es im Rahmen des hier entworfenen Konzepts möglich die Verknüpfungsprozesse eines MPI auf anonymen Daten durchzuführen was die Einhaltung des Datenschutzes auch in einem verteilten Umfeld erlaubt. Des Weiteren kann durch den Einsatz von Maklerknoten, in denen Kataloge mit anonymen Datensätzen geführt werden, eine effiziente Suche gewährleistet werden. Bei der Auswahl der zu kodierenden Attributskombinationen muss ein Kompromiss gefunden werden, zwischen dem höheren Rechen- und Speicheraufwand bei Mitführung von mehr Hashwerten und der Qualität der Suche auf den fertig anonymisierten Datensätzen. Hierbei gilt, dass bei sehr hoher Qualität der Ursprungsdaten bereits geringe Kombinationsmöglichkeiten ausreichen.

# Abbildungsverzeichnis

4.1. Process Flow with Patient Identifier Cross-referencing . . . . . 8
4.2. PIX - Transaktionen . . . . . 12
4.3. PIX Query . . . . . 14
4.4. Domain Reference Model for PIDS . . . . . 17
4.5. PIDS Components and Inheritence Diagram . . . . . 18
4.6. MPI System mit PIDS Interface . . . . . 22

6.1. Verteilte Referenzierung . . . . . 35
6.2. Lokale Durchführung der Vergleichsoperationen . . . . . 44
6.3. Verteilte Durchführung der Vergleichsoperation . . . . . 45
6.4. Einsatz von Maklern in der Suche . . . . . 47
6.5. Use-Case des Suchvorgangs . . . . . 49

# Tabellenverzeichnis

4.1. Conformance Classes . . . . . 21
4.2. Attribute für Vergleichsoperationen . . . . . 27

5.1. Soundex Kodierung . . . . . 30
5.2. Kodierung gemäß dem Kölner Verfahren . . . . . 32

A.1. PID Segment HL7v2 . . . . . 64

# Anhang A.

| SEQ | LEN | DT | OPT | RP/# | TBL# | ITEM# | ELEMENT NAME |
|---|---|---|---|---|---|---|---|
| 1 | 4 | SI | O | | | 00104 | Set ID - Patient ID |
| 2 | 20 | CX | O | | | 00105 | Patient ID (External ID) |
| 3 | 20 | CX | R | Y | | 00106 | Patient ID (Internal ID) |
| 4 | 20 | CX | O | Y | | 00107 | Alternate Patient ID - PID |
| 5 | 48 | XPN | R | | | 00108 | Patient Name |
| 6 | 48 | XPN | O | | | 00109 | Mother's Maiden Name |
| 7 | 26 | TS | O | | | 00110 | Date/Time of Birth |
| 8 | 1 | IS | O | | 0001 | 00111 | Sex |
| 9 | 48 | XPN | O | Y | | 00112 | Patient Alias |
| 10 | 1 | IS | O | | 0005 | 00113 | Race |
| 11 | 106 | XAD | O | Y | | 00114 | Patient Address |
| 12 | 4 | IS | B | | | 00115 | County Code |
| 13 | 40 | XTN | O | Y | | 00116 | Phone Number - Home |
| 14 | 40 | XTN | O | Y | | 00117 | Phone Number - Business |
| 15 | 60 | CE | O | | 0296 | 00118 | Primary Language |
| 16 | 1 | IS | O | | 0002 | 00119 | Marital Status |
| 17 | 3 | IS | O | | 0006 | 00120 | Religion |
| 18 | 20 | CX | O | | | 00121 | Patient Account Number |
| 19 | 16 | ST | O | | | 00122 | SSN Number - Patient |
| 20 | 25 | CM | O | | 6 | 00123 | Driver's License Number - Patient |
| 21 | 20 | CX | O | Y | | 00124 | Mother's Identifier |
| 22 | 3 | IS | O | | 0189 | 00125 | Ethnic Group |
| 23 | 60 | ST | O | | | 00126 | Birth Place |
| 24 | 2 | ID | O | | 0136 | 00127 | Multiple Birth Indicator |
| 25 | 2 | NM | O | | | 00128 | Birth Order |
| 26 | 4 | IS | O | Y | 0171 | 00129 | Citizenship |
| 27 | 60 | CE | O | | 0172 | 00130 | Veterans Military Status |
| 28 | 80 | CE | O | | | 00739 | Nationality |
| 29 | 26 | TS | O | | | 00740 | Patient Death Date and Time |
| 30 | 1 | ID | O | | 0136 | 00741 | Patient Death Indicator |

Tabelle A.1.: PID Segment HL7v2

# Literaturverzeichnis


[ACC07a] ACC, HIMSS and RSNA. IT Infrastructure Technical Framework Volume 1 - Integration Profiles. Technical report, Integrating the Healthcare Enterprise, August 2007. Revision 4.0 - Final Text.

[ACC07b] ACC, HIMSS and RSNA. IT Infrastructure Technical Framework Volume 2 - Transactions. Technical report, Integrating the Healthcare Enterprise, August 2007. Revision 4.0 - Final Text.

[AST08] Maarten van Steen Andrew S. Tanenbaum. *Verteilte Systeme*, volume 2. Pearson Studium, 2008.

[Bla07] Paul E. Black. Dictionary of Algorithms and Data Structures. `http://www.itl.nist.gov/div897/sqg/dads/HTML/metaphone.html`, July 2007. last Visited at: 23.06.09.

[Bla09] Paul E. Black. Dictionary of Algorithms and Data Structures. `http://www.itl.nist.gov/div897/sqg/dads/HTML/nysiis.html`, März 2009. last Visited at: 23.06.09.

[CB02] Mark R. Chassin and Elise C. Becher. The Wrong Patient. *Ann Intern Med.*, 136:826–833, 2002. Academia and Clinic Quality Grand Rounds.

[DLAE07] A. Dogac, G.B. Laleci, T. Aden, and M Eichelberg. Enhancing IHE XDS for Federated Clinical Affinity Domain Support. *IEEE Transactions on Information Technology in Biomedicine*, Vol. 11:213–221, March 2007.

[ED01] Jones P. Eastlake D. RFC 3174 - US Secure Hash Algorithm 1 (SHA1). `http://tools.ietf.org/html/rfc3174`, September 2001.

[GEvS07] P. Garbacki, D. H. J. Epema, and M. van Steen. Optimizing Peer Relationships in a Super-Peer Network. In *27th International Conference on Distributed Computing Systems (ICDCS 2007)*, Toronto, Canada, June 2007.

[Lev66] Vladimir Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10:707–710, 1966.

[Mem01] Klinikum Memmingen. Der promedtheus Master Patient Index am Klinikum Memmingen. http://www.promedtheus.de/opencms/export/sites/promedtheus/download/anwenderbericht.pdf, 2001. last downloaded at: 18.06.09.

[Obj01] Object Management Group Inc. Person Identification Service (PIDS) Specification. http://www.omg.org/cgi-bin/doc?formal/01-04-04.pdf, April 2001. last downloaded at: 18.06.09.

[Pos69] Hans Joachim Postel. Die Kölner Phonetik. Ein Verfahren zur Identifizierung von Personennamen auf der Grundlage der Gestaltanalyse. *IBM-Nachrichten*, 19. Jahrgang:S. 925–931, 1969.

[Rep07] Dominic John Repici. Understanding Classic SoundEx Algorithms. http://www.creativyst.com/Doc/Articles/SoundEx1/SoundEx1.htm last Visited at: 21.06.09, 2007.

[Riv92] R. Rivest. RFC 1321 - The MD5 Message-Digest Algorithm. http://tools.ietf.org/html/rfc1321, April 1992.

[Sun08a] Sun Microsystems. About Sun Master Patient Index. http://developers.sun.com/docs/javacaps/designing/jcapsdeindxspvw.dsgn_mpi-about_c.html, December 2008. last visited at: 18.06.09.

[Sun08b] Sun Microsystems. IHE Compliant Master Patient Index based on Sun JavaTM Composite Application Platform Suite (CAPS). http://www.sun.com/solutions/landing/industry/healthcare/ihe_mpi/ihe_javacaps.pdf, February 2008. last visited at: 18.06.09.

[Sun08c] Sun Microsystems. Sun Master Patient Index Overview. http://developers.sun.com/docs/javacaps/designing/jcapsdeindxspvw.ghbdc.html, December 2008. last visited at: 18.06.09.

[TAS94] W. Thoben, H.-J. Appelrath, and S. Sauer. Record Linkage of Anonymous Data by Control Numbers. In W. Gaul & D. Pfeifer, editor, *From Data to Knowledge: Theoretical and Practical Aspects of Classification, Data Analysis and Knowledge Organisation*, pages 412–419, Oldenburg, 1994. Springer Verlag.

[WD01] Jörg Wünnemann and Carl Dujat. Die Rolle eines Master Patient Index in verteilten Informationssystemen. *Telemedizinführer Deutschland*, Ausgabe 2001:156–161, 2001.

[Wil05] Martin Wilz. Aspekte der Kodierung phonetischer Ähnlichkeiten in deutschen Eigennamen. Magisterarbeit, Universität zu Köln, Philosophische Fakultät, 2005.