Institut für Informatik
der
Technischen Universität München

CSCW in der Bioinformatik: Ein objektorientiertes Groupwaresystem zur Unterstützung in der Gen- und Genomanalyse

Andreas Kaps

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr. M. Broy
Prüfer der Dissertation:	1.	Univ.-Prof. Dr. J. Schlichter
	2.	Univ.-Prof. Dr. H.W. Mewes, Ludwig-Maximilians-Universität München

Die Dissertation wurde am 29.1.2001 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 7.6.2001 angenommen.

Zusammenfassung

In der modernen Biologie ist es möglich geworden, Zusammenhänge auf molekularer Ebene zu untersuchen. Molekularbiologische Experimente generieren heute Daten, deren Organisation und Auswertung nur durch moderne Informationstechnologie möglich ist. Es hat sich ein neues Teilgebiet der Informatik entwickelt, die Bioinformatik.

War zu Beginn der rechnergestützten Sequenzdatenanalyse eine manuelle Bearbeitung aller bekannten Proteinsequenzen am Rechner möglich, ist die Erfassung, Analyse, Präsentation und Wartung dieser Datensammlungen heutzutage nurmehr mit Rechnerunterstützung durchführbar. Die biomedizinische Forschung beispielsweise generiert notwendige Basisinformationen, ohne die Anwendungen sowohl in der roten (Medizin, Pharmazie) als auch in der grünen (Agrar) Biotechnologie nicht möglich wären.

Neben etablierten Anwendungsgebieten der Informatik in der Biologie, wie z.B. Algorithmen zum Sequenzvergleich, Datenbanken oder Neuronale Netze, ist die Unterstützung der Gruppenkommunikation durch ein CSCW-System bisher nicht untersucht worden. Die Zusammenarbeit von Wissenschaftlern im Bereich molekularbiologischer Sequenzdatenanalyse wurde daher in dieser Arbeit unter dem Aspekt der rechnergestützten Gruppenarbeit untersucht. Es wurden zunächst Gruppen der wissenschaftlichen Arbeitsgruppe *MIPS (Münchener Informationszentrum für Proteinsequenzen)* am Max-Planck-Institut für Biochemie, Martinsried, unter der Leitung von Herrn Prof. Dr. H.-W. Mewes, definiert, sowie ihre Aufgaben und Arbeitsweisen analysiert. Bei dieser Analyse aufgetretene Unzulänglichkeiten in der Rechnerunterstützung schufen die Grundlage, einen Anforderungskatalog für ein CSCW-System formulieren zu können. Darauf aufbauend wurde ein Groupwaresystem entworfen, das eine bessere Unterstützung der Gruppenmitglieder zum Ziel hatte. Der Entwurf wurde prototypisch implementiert und in der wissenschaftlichen Arbeitsgruppe MIPS eingesetzt.

Abschließend wurde unter den beteiligten Gruppenmitgliedern eine schriftliche Umfrage durchgeführt. Ziel dieser empirischen Untersuchung war die Ermittlung der Akzeptanz des Systems, das sich im täglichen Einsatz befand. Zum Zeitpunkt der Umfrage waren Teile des Systems zwischen sechs Monate und drei Jahre verfügbar.

Das Ergebnis dieser Umfrage ergab eindeutig, daß sich die alltägliche Arbeit der Gruppenmitglieder durch das System deutlich verbessert hat.

Danksagung

Diese Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Münchener Informationszentrum für Proteinsequenzen (MIPS) am Max-Planck-Institut für Biochemie, Martinsried.

Zuallererst möchte ich Herrn Prof. H.-Werner Mewes ganz herzlich für die Überlassung dieses Themas danken. Er hat mir den notwendigen Freiraum für die Bearbeitung ermöglicht und war offen für spontane Diskussionen (auch wenn diese gelegentlich in Gespräche über Literatur oder Jazz umkippten).

Herrn Prof. Johann Schlichter möchte ich danken für seine Bereitschaft, mich als externen Doktoranden zu betreuen. Seine Kritik und Anregungen in unseren regelmäßigen Treffen sowie seine Anmerkungen in Fassungen der schriftlichen Ausarbeitung waren mir eine große Hilfe.

Allen MitarbeiterInnen der Arbeitsgruppe MIPS sei herzlich gedankt. Besonders Frau Dr. Susanne Stocker und Herrn Dr. Friedhelm Pfeiffer, die ich mit so manchen biologischen Fragen gelöchert habe. Herr Dipl.-Inform. Dirk Haase hat Teile der Arbeit implementiert. Vielen Dank dafür. Herrn Dipl.-Inform. Normann Strack sei gedankt für seine konstruktiven Ratschläge im Zusammenhang mit *ObjectStore* und C++. Herrn Dr. Manfred Gerstner danke ich für die umfassenden Diskussionen zum Thema objektorientiertes Design sowie für unsere sonstigen geistreichen Gespräche.

Ein ganz besonderes Dankeschön geht an Herrn Andreas Vlasic vom Medieninstitut in Ludwigshafen, der mir zu Fragen der empirischen Sozialforschung wertvolle Informationen gegeben hat (und der Rotwein war ausgezeichnet).

Ein herzliches Dankeschön meinen jetzigen Kolleginnen und Kollegen der Biomax Informatics AG, die mich in der Endphase durch ihr gutgemeintes „Na, wie geht's der Doktorarbeit“ immer wieder ermutigt haben.

Nicht zuletzt sei meiner Frau, Frau Dr. med. Annette Kaps, herzlichst gedankt. Sie hat die Arbeit nicht nur Korrektur gelesen, sondern sie hat mich auch immer zur rechten Zeit an den Rechner geschickt bzw. vom Rechner weggeholt.

Inhaltsverzeichnis

1 Einleitung **7**
- 1.1 Einführung . . . 7
  - 1.1.1 Proteine als Bausteine des Lebens . . . 8
  - 1.1.2 Notwendigkeit der Rechnerunterstützung . . . 9
- 1.2 Ziel der Arbeit . . . 9
- 1.3 Aufbau der Arbeit . . . 10

2 Hintergrund: Molekulare Sequenzdatenanalyse **12**
- 2.1 Biologischer Hintergrund . . . 12
  - 2.1.1 Aminosäuren und Proteine . . . 13
  - 2.1.2 Speicherung genetischer Information im Organismus . . . 14
  - 2.1.3 Genexpression und Proteinbiosynthese . . . 15
- 2.2 Genomsequenzierung: Vom Organismus zur genetischen Information . . . 17
- 2.3 Sequenzvergleich als Analysemethode . . . 19
- 2.4 Molekularbiologische Datensammlungen . . . 20
  - 2.4.1 Nukleinsäuredatenbanken . . . 21
  - 2.4.2 Proteinsequenzdatenbanken . . . 23
  - 2.4.3 Spezialisierte Datenbanken . . . 26

3 CSCW in der Biologie: Anforderungen und existierende Ansätze **27**
- 3.1 Das Forschungsgebiet *Rechnergestützte Gruppenarbeit* . . . 27
  - 3.1.1 *Computer Supported Cooperative Work* - CSCW . . . 27
  - 3.1.2 CSCW und Groupware . . . 29
  - 3.1.3 Interdisziplinarität von CSCW . . . 31
  - 3.1.4 Klassifizierung von CSCW-Systemen . . . 32
- 3.2 Gruppen in der Gen- und Genomanalyse . . . 34
  - 3.2.1 Definitionen . . . 34
  - 3.2.2 Wartung einer Proteinsequenzdatenbank . . . 35
  - 3.2.3 Systematische Genomsequenzierungsprojekte . . . 40
  - 3.2.4 Analyse kompletter Genome . . . 43

3.2.5 Systematische Funktionsanalyseprojekte 44
3.3 Anwendungsszenarien 46
3.3.1 Die wissenschaftliche Arbeitsgruppe MIPS 47
3.3.2 Wartung einer Proteinsequenzdatenbank 48
3.3.3 Literaturverwaltung 54
3.4 Anforderungen an ein CSCW-System 57
3.4.1 Allgemeine Anforderungen 58
3.4.2 Spezielle Anforderungen 62
3.5 Existierende Ansätze 64
3.5.1 Systeme zur molekularen Sequenzdatenanalyse 64
3.5.2 Diskussion 67

4 Entwurf eines Groupwaresystems für die Gen- und Genomanalyse 69
4.1 Datenhaltung 69
4.1.1 Verteilte Datenhaltung 70
4.1.2 Fehlermöglichkeiten in verteilten Systemen 71
4.1.3 Eingesetzte Strategie der verteilten Datenhaltung 72
4.2 Nebenläufigkeitskontrolle 72
4.2.1 Allgemeine Betrachtung 72
4.2.2 Eingesetzte Strategien zur Nebenläufigkeitskontrolle . . . 73
4.3 Datenverwaltung 74
4.3.1 Reduzierung der Komplexität 74
4.3.2 Datenbankkomponente 75
4.4 Datenzugriff 77
4.4.1 Kommunikationsschicht 78
4.4.2 Der Standard *CORBA* 81
4.5 Infrastruktur des Groupwaresystems 82
4.6 Automatisierter Datenimport 83
4.7 Awareness 87
4.8 Ausgewählte Applikationen 89
4.8.1 Zentrale Literaturverwaltung 89
4.8.2 Spezialisierte Literaturverwaltung 100
4.8.3 Management von Proteinsequenzen 102
4.8.4 Systematische Genomsequenzierungsprojekte 106

5 Realisierung des entworfenen Groupwaresystems 111
5.1 Datenbankkomponente 111
5.1.1 Persistenter Speicher 112
5.1.2 Verwaltungsinstanz 113
5.1.3 Kontrollinstanz 116
5.1.4 Zugriffsdienst 120

5.2 Kommunikationsschicht 120
5.2.1 CORBA 121
5.2.2 RPC 122
5.3 Informationswandler 125
5.3.1 Importeinheit 125
5.3.2 Aufbereitungseinheit 126
5.3.3 Integrationseinheit 127
5.3.4 Manuelle Bearbeitungen 128
5.4 Awareness 129
5.5 Ausgewählte Applikationen 130
5.5.1 Literaturverwaltung 130
5.5.2 Management von Proteinsequenzen 134
5.5.3 Genomsequenzierungsprojekte 142

6 Einsatz und empirische Akzeptanzanalyse des prototypischen Groupwaresystems 145
6.1 Einsatz des prototypischen Groupwaresystems 145
6.2 Datenerhebung 146
6.2.1 Empirische Sozialforschung 146
6.2.2 Datenerhebungstechniken 147
6.2.3 Theoretische Grundlagen für Befragungen 148
6.2.4 Eingesetzte Befragungsart 153
6.3 Hypothesen 154
6.4 Durchgeführte Befragung 154
6.5 Ergebnisse der Befragung 155
6.5.1 Einleitungsfragen 1 bis 5 155
6.5.2 Sach- und Kontrollfragen 6a bis 12b 156
6.5.3 Fragen zur Person 13 bis 22 159
6.6 Diskussion der Ergebnisse 160
6.6.1 Gruppenzusammensetzung 160
6.6.2 Bewertung der Hypothesen 161
6.6.3 Zusammenfassung 163

7 Zusammenfassung und Ausblick 164
7.1 Zusammenfassung 164
7.2 Ausblick 166

A Fragebogen Groupware 170

Kapitel 1

Einleitung

1.1 Einführung

In der modernen Biologie ist es möglich geworden, Zusammenhänge auf molekularer Ebene zu untersuchen. Molekularbiologische Experimente generieren heute Daten, deren Organisation und Auswertung nur durch moderne Informationstechnologie möglich ist. Es hat sich ein neues Teilgebiet der Informatik entwickelt, die Bioinformatik. Der Bundesminister für Forschung und Technologie (BMFT) hat 1992 ein Strategiekonzept *Molekulare Bioinformatik* präsentiert. Darin werden Informatiker aufgerufen, „gemeinsam mit Molekularbiologen, Biochemikern und Pharmazeuten rechnergestützte Methoden für die Analyse biologischer Substanzen zu entwickeln“ (zitiert nach [Len96]). 1999 hat die Deutsche Forschungsgemeinschaft (DFG) eine *Initiative Bioinformatik* ausgeschrieben, im Rahmen derer 50 Millionen DM zur Förderung des interdisziplinären Forschungsgebietes Bioinformatik bereitstehen. Mit dieser Initiative sind Universitäten aufgerufen, „in regionalen Kooperationen und im Verbund mit außeruniversitären Einrichtungen Konzepte für die Etablierung oder den Ausbau eines eigenen Wissenschafts- und Ausbildungsprofils im Bereich der Bioinformatik an ihrem Standort vorzulegen“.[1]

Das in den letzten Jahren erworbene Wissen in den Biowissenschaften ermöglicht die industrielle Nutzung. Der Bereich der Biotechnologie hat sich zu einem schnell wachsenden und wichtigen Wirtschaftssektor entwickelt. Die Anwendungen sind vielschichtig und reichen von der medizinischen Diagnostik und Therapie über die Erzeugung von Pathogenresistenzen in Kulturpflanzen bis hin zur Entwicklung von Klebstoffen und Waschmitteln.

Die Fortschritte in der biomedizinischen Forschung wecken Hoffnungen, eine unmittelbare Behandlung von Erkrankungen auf molekularer Ebene durchführen zu können. Grundvoraussetzung ist das Wissen um die wichtigsten Bestandteile

[1] Ausschreibung Initiative Bioinformatik, siehe http://www.dfg.de

der Zelle, der elementaren biologischen Organisationsform lebender Organismen.

1.1.1 Proteine als Bausteine des Lebens

Die wichtigsten Makromoleküle lebender Organismen sind Proteine, die aus zwanzig verschiedenen Bausteinen, den Aminosäuren, aufgebaut sind. Die Informationen über den Aufbau der Proteine sind im Genom kodiert. Das *zentrale Dogma* der Molekularbiologie besagt, daß der gesamte Bauplan einer Zelle in der DNS enthalten ist. In der lebenden Zelle werden diese Baupläne in mRNS transkribiert, an den Ort der Proteinbiosynthese transportiert und dort in die Proteinsequenz translatiert.

Die genetische Information eines Organismus ist eine notwendige Informationsquelle für Aufbau und Vermehrung des Individuums. Der Aufbau eines Genoms entspricht seiner zellulären Komplexität: bakterielle Genome sind kleiner und leichter zu interpretieren als Genome hochdifferenzierter Organismen ([MPH97]). Genome hochdifferenzierter Organismen können um den Faktor 1000 größer sein als etwa die von Bakterien.

Zur Bestimmung des Aufbaus eines Proteins existiert als klassische Methode der Biochemie die aufwendige Proteinsequenzierung. Diese kostspielige und nicht immer erfolgreich durchführbare Methode wurde aufgrund der in den 80er Jahren neu entwickelten molekularbiologischen Techniken durch die DNS-Sequenzierung ersetzt. Statt direkt das Protein zu analysieren, wird bei dieser Technologie der Bauplan des Proteins im Genom bestimmt. Diese Techniken ermöglichen eine schnelle und kostengünstige Bestimmung kompletter Genome. Als Ergebnis systematischer Genomsequenzierungsprojekte liegen Sequenzdaten vor, die nicht nur interessante, für Proteine kodierende Bereiche enthalten, sondern auch Teile des Genoms, über deren Bedeutung (noch) kein Wissen vorhanden ist. Experimentelle Daten systematischer Genomanalysen müssen von Wissenschaftlern bearbeitet und analysiert werden: die Aufgaben reichen von der Erstellung genetischer, physikalischer und molekularer Karten bis zur Identifizierung und Charakterisierung von genetischen Elementen. Besonderes Gewicht liegt auf der funktionellen Analyse mit dem Ziel, Funktionen genetischer Elemente, vor allem die Interaktionen, die ihre zelluläre Rolle bestimmen, aufzuklären.

Systematische Genomanalysen sind in erster Linie datenorientiert. Die Arbeit am Rechner, die, neben der Sammlung, aus der Analyse der durch molekularbiologische Verfahren ermittelten Daten besteht, gewann in den letzten Jahren zunehmend an Bedeutung. Ursache hierfür sind stark anwachsende Datenmengen aufgrund systematischer Sequenzierungen kompletter Genome. Seit ca. zehn Jahren werden umfassende Sequenzierungsprojekte durchgeführt, oftmals von Konsortien. Als Folge davon werden in immer kürzeren Abständen gesamte Genome in wissenschaftlichen Journalen publiziert (die ersten komplett bestimmten Genome

wurden in [FAW+95], [FGW+95], [BWO+96], [GBB+96], [TWK+97] beschrieben). Aufgrund der verbesserten Sequenzierungstechniken und der Zunahme der Zahl systematischer Sequenzierungsprojekte verdoppelt sich der Umfang biologischer Sequenzdatensammlungen gegenwärtig alle zweieinhalb Jahre. Die Nukleinsäuredatenbank *EMBL Nucleotide Sequence Database* enhält beispielsweise mehr als drei Millionen Einträge mit mehr als zwei Milliarden Basenpaaren (Release 58, März 1999).

1.1.2 Notwendigkeit der Rechnerunterstützung

War zu Beginn der rechnergestützten Sequenzdatenanalyse eine manuelle Bearbeitung aller bekannten Proteinsequenzen am Rechner möglich, ist die Erfassung, Analyse, Präsentation und Wartung dieser Datensammlungen heutzutage nur mehr mit Rechnerunterstützung durchführbar. Stark anwachsende Datenmengen müssen von Mitarbeitern in Forschungsinstituten bearbeitet werden. Da öffentlich geförderte Einrichtungen der Grundlagenforschung nicht über ausreichende finanzielle Mittel verfügen, Vollständigkeit und hohe Qualität der Datensammlungen andererseits essentielle Grundlagen für eine effiziente Forschung sind, muß die Effizienz einer in der Sequenzdatenanalyse tätigen Gruppe entsprechend erhöht werden. Die biomedizinische Forschung generiert notwendige Basisinformationen, ohne die Anwendungen sowohl in der roten (Medizin, Pharmazie) als auch in der grünen (Agrar) Biotechnologie nicht möglich wären.

Die Forderung nach hoher Qualität bei höherem Durchsatz kann nur durch entsprechenden Einsatz von spezialisierter Software realisiert werden.

1.2 Ziel der Arbeit

Neben etablierten Anwendungsgebieten der Informatik in der Biologie, wie z.B. Algorithmen zum Sequenzvergleich, Datenbanken, Neuronale Netze oder Hidden Markov Models, ist die Unterstützung der Gruppenkommunikation durch ein CSCW-System bisher nicht untersucht worden. Um eine optimale Unterstützung der Bioinformatiker, die in der molekularbiologischen Grundlagenforschung tätig sind, durch Rechner ermöglichen zu können, wurde ein Groupwaresystem entworfen, prototypisch realisiert und in einer wissenschaftlichen Gruppe eingesetzt.

Die Zusammenarbeit von Wissenschaftlern im Bereich molekularbiologischer Sequenzdatenanalyse wird in dieser Arbeit unter dem Aspekt der rechnergestützten Gruppenarbeit untersucht. Es wurden zunächst Gruppen der wissenschaftlichen Arbeitsgruppe *MIPS (Münchener Informationszentrum für Proteinsequenzen)*, die in diesem Teilgebiet der biologischen Grundlagenforschung tätig ist, definiert, sowie ihre Aufgaben und Arbeitsweisen analysiert. Ein Fokus wurde

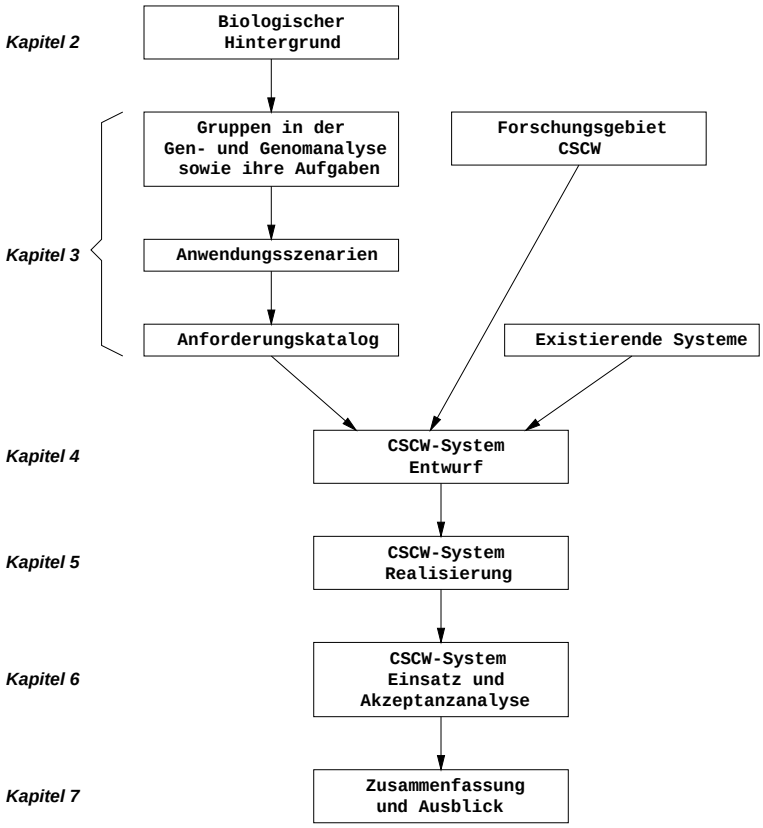

Abbildung 1.1: Graphische Darstellung des Aufbaus der Arbeit.

dabei auf eine Untergruppe, die Annotationsgruppe der Proteinsequenzdatenbank *PIR-International* gerichtet. Bei dieser Analyse aufgetretene Unzulänglichkeiten in der Rechnerunterstützung schufen die Grundlage, um einen Anforderungskatalog für ein CSCW-System formulieren zu können. Darauf aufbauend wurde ein Groupwaresystem entworfen, das eine bessere Unterstützung der Gruppenmitglieder zum Ziel hatte. Der Entwurf wurde prototypisch implementiert und in der Arbeitsgruppe MIPS eingesetzt. Anschließend erfolgte durch eine Befragung der Gruppenmitglieder eine empirische Analyse der Akzeptanz des prototypischen Systems.

1.3 Aufbau der Arbeit

In *Kapitel 2* wird eine kurze Einführung in das Anwendungsgebiet *Molekulare Sequenzdatenanalyse* gegeben. Es werden biologische Grundbegriffe erläutert. Wei-

terhin werden Methoden der Sequenzdatenanalyse sowie molekularbiologische Datensammlungen vorgestellt, sofern sie für die Arbeit von Bedeutung waren. Kapitel 2 dient als Grundlage für die Darstellung von Anwendungsszenarien in Kapitel 3, aus denen Anforderungen an ein CSCW-System abgeleitet werden.

In *Kapitel 3* wird kurz in den Informatik-Forschungsbereich CSCW eingeführt. Anschließend werden Gruppen vorgestellt, die in der Gen- und Genomanalyse tätig sind. Für diese Gruppen werden charakteristische Anwendungsszenarien dargestellt, aus denen Anforderungen an ein CSCW-System abgeleitet werden. Es folgt die Darstellung und Bewertung bereits existierender Ansätze für diesen Bereich. Dazu werden in der wissenschaftlichen Literatur publizierte Systeme unter dem Gesichtspunkt der Gruppenunterstützung erstmals analysiert. Eine Zusammenfassung der wichtigsten Ergebnisse schließt das Kapitel ab.

In *Kapitel 4* wird ein Groupwaresystem entworfen. Es wird eine Infrastruktur dargestellt, die als Grundlage für die entwickelten prototypischen Applikationen des CSCW-Systems diente. Die dem System zugrundeliegende flexible Softwarearchitektur wird vorgestellt. Schließlich werden aus den Anforderungen exemplarisch Applikationen herausgegriffen, für die ein Konzept entworfen wurden.

Kapitel 5 beschäftigt sich mit Implementierungsaspekten des CSCW-Systems. Es wird auf die prototypische Realisierung von Applikationen im CSCW-System eingegangen.

Kapitel 6 behandelt die durchgeführte Akteptanzanalyse des entwickelten Systems, das in der Arbeitsgruppe MIPS im täglichen Einsatz ist. Es wurde unter den Anwendern eine Umfrage durchgeführt, durch die die Akzeptanz des CSCW-Systems ermittelt werden sollte. Es werden Hypothesen formuliert, die durch die Befragung bestätigt oder auch widerlegt werden sollten. Nach Darstellung möglicher Datenerhebungstechniken der empirischen Sozialforschung wird die eingesetzte Methode vorgestellt. Die erzielten Ergebnisse werden aufgeführt und im Hinblick auf die Hypothesen analysiert und diskutiert. Der entwickelte Fragebogen ist im Anhang abgedruckt.

Im abschließenden Kapitel werden das entwickelte Konzept sowie die Erfahrungen mit dem eingesetzten Groupwaresystem zusammengefaßt. Ein sich daraus ergebender Ausblick schließt die Arbeit ab.

Kapitel 2

Hintergrund: Molekulare Sequenzdatenanalyse

In diesem Kapitel werden nach Darstellung des biologischen Hintergrundes Arbeitsweisen und Methoden in der rechnergestützten molekularbiologischen Sequenzdatenanalyse vorgestellt. Dazu werden Analysekonzepte sowie die dazu notwendigen Datensammlungen beschrieben. Es soll ein Eindruck der inhärenten Komplexität biologischer Annotationen und ihrer Verwaltung vermittelt werden. Es werden nur Bereiche betrachtet, die für das zu entwickelnde CSCW-System von Bedeutung waren. Ziel dieses Kapitels ist es, eine Grundlage zu schaffen, auf der im anschließenden Kapitel ein Anforderungsprofil für ein Groupwaresystem im Bereich der molekularen Sequenzdatenanalyse beschrieben werden kann.

2.1 Biologischer Hintergrund

Die Aufklärung der Funktion biologischer Makromoleküle durch Analyse der genetischen Informationen lebender Systeme ist das primäre Ziel der Gen- und Genomanalyse. Das *Genom* bezeichnet die Gesamtheit aller genetischen Informationen eines Organismus. Es kann als sequentieller Datenspeicher angesehen werden, der Informationen über den molekularen Aufbau und die Funktion der Zellen enthält. Ein Teil des Genoms enthält für Proteine kodierende Bereiche. Daneben gibt es Bereiche, die z.B. regulatorische Aufgaben besitzen. Nach derzeitigem Kenntnisstand existieren außerdem Abschnitte, die keine Bedeutung besitzen oder deren Bedeutung noch nicht ermittelt wurde.

Von zentralem Interesse ist das Finden kodierender Bereiche im Genom eines Organismus sowie die Funktionsanalyse der korrespondierenden Proteine in lebenden Zellen. Je komplexer ein Organismus aufgebaut ist, desto ungünstiger ist im allgemeinen das Verhältnis aus kodierenden zu nicht-kodierenden Berei-

chen im Genom. In höher organisierten Lebewesen sind zudem kodierende Abschnitte von langen nicht-kodierenden Bereichen unterbrochen, was die Suche nach kodierenden DNS-Sequenzen weiter erschwert. Das Wissen um regulatorische Aspekte, d.h. zu welchem Zeitpunkt im Lebenszyklus einer Zelle oder bei welchen Umwelteinflüssen welche Mengen einzelner Proteine produziert werden, gehört zu den Grundvoraussetzungen eines profunden Verständnisses des zellulären Geschehens.

2.1.1 Aminosäuren und Proteine

Proteine sind die komplexesten Makromoleküle lebender Organismen. Ihre Aufgaben sind vielfältig. Wichtige Funktionen erfüllen sie als Strukturbestandteile, wie z.B. die Proteine des Cytoskeletts.[1] Enzyme ermöglichen, beschleunigen oder kontrollieren chemische Reaktionen in lebenden Systemen ohne dabei selbst in der Reaktion verbraucht zu werden. Immunglobuline, Vermittler biologischer Abwehrmechanismen, sind ebenfalls Proteine.

Alle Proteine, seien sie für Struktur, Katalyse oder Erkennung verantwortlich, sind aus wenigen gleichen Bausteinen aufgebaut. In den bisher bekannten Proteinen finden sich 20 verschiedene *proteinogene Aminosäuren*.[2]

Primärstruktur Proteine sind Polymerisate der 20 proteinogenen Aminosäuren. Die Sequenz der einzelnen Aminosäuren wird als *Primärstruktur* bezeichnet. Diese Ketten sind nicht verzweigt, sondern mit einer Perlenkette vergleichbar.[3]

Sekundärstruktur Vermessungen von Proteinen mit Hilfe der Röntgenstrukturanalyse ergaben, daß die vorhandenen Atomabstände nicht mit der Annahme übereinstimmen, die Polypeptidkette sei vollständig gestreckt. Die Entdeckung der α-Helix und β-Faltblattstruktur, die die wichtigsten *Sekundärstrukturelemente* eines Proteins bilden, klärte diesen Widerspruch auf.

Tertiärstruktur Der Polypeptidstrang liegt in der lebenden Zelle als Knäuel vor. Diese räumliche Struktur eines Proteins ist essentiell für seine Funktion in der

[1] Das genau strukturierte, dynamisch organisierte Cytoskelettsystem eukaryoter Zellen, d.h. Zellen mit Zellkern, ist an wichtigen Funktionen innerhalb der Zelle beteiligt: Zellteilung, Zellmobilität, Formerhaltung von Zellen, Polarität von Zellen.

[2] In der Natur kommen mehr als 20 Aminosäuren vor. Die nichtproteinogene Aminosäure Citrullin z.B. ist ein wichtiges Zwischenprodukt im Harnstoffzyklus, ein Stoffwechselvorgang, im dem das für den Organismus giftige Ammoniak in Form des Harnstoffs fixiert und somit entgiftet wird.

[3] Enthält ein Protein weniger als 100 Aminosäuren, spricht man von *Peptiden*. In dieser Arbeit wird *Protein* als Sammelbegriff für Peptide und Proteine verwendet.

lebenden Zelle. Im Knäuel kommen sich Bereiche des Peptidstranges so nahe, daß Wechselwirkungen nicht benachbarter Aminosäuren hochorganisierte Strukturen stabilisieren können (*Tertiärstruktur*).

Kleine Schädigungen der genetischen Information, z.B. durch eine Punktmutation, bei der eine einzelne Aminosäure durch eine andere Aminosäure ausgetauscht wurde, oder der Verlust eines Nukleotids, können schwerwiegende Beeinträchtigungen der Raumstruktur eines Proteins zur Folge haben. Mutationen sind die Ursachen genetischer Erkrankungen. Im Fall strukturverändernder Mutationen kann das Protein seine Funktion in der Zelle nicht mehr ausüben. Funktionsverluste treten selten vollständig auf (*dominante Mutanten*), da der doppelte Chromosomensatz (z.B. beim Menschen) gravierende Defekte verhindert. Die Kombination rezessiver Defekte kann jedoch zu Funktionsverlusten führen.

Quartärstruktur Lagern sich mehrere Peptide als Untereinheiten zu einer Funktionseinheit zusammen, spricht man von der *Quartärstruktur* eines Proteins.

2.1.2 Speicherung genetischer Information im Organismus

Der Informationsspeicher genetischer Information ist die *Desoxyribonukleinsäure (DNS)*. Die Entdeckung der DNS-Konformation durch James Watson und Francis Crick im Jahre 1953 ist sicherlich eine der bedeutensten Entdeckungen der Biochemie des 20. Jahrhunderts, die 1962 mit dem Nobelpreis für Medizin ausgezeichnet wurde. Ausgangspunkt für die Genetik im allgemeinen waren Beobachtungen des Augustinermönchs Gregor Mendel,[4] der bei seinen *Versuchen über Pflanzenhybriden* (1865) Studien zur Vererbung von Blütenfarben und Fruchtformen bei der Erbse anstellte. Er erkannte, daß es Gesetzmäßigkeiten bei der Vererbung gibt. Diese als Mendelsche Regeln bekannt gewordenen Gesetze bildeten die Grundlage der experimentellen Genetik.

Das Genom eines jeden lebenden Organismus besitzt die von Mendel erkannten Faktoren, heute *Gene* genannt. Die Anzahl schwankt jedoch stark: das HIV-Virus enthält 9 Gene, die Bäckerhefe ca. 6.000, Pflanzen ca. 25.000. Bei Maus und Mensch vermutet man 80.000 bis 140.000 Gene. Rechnet man allerdings Splicevarianten[5] und post-translatante Prozessierungen hinzu, dürften weit mehr Proteine in menschlichen Zellen vorkommen.

Die DNS ist bei allen Organismen sowohl strukturell als auch chemisch gleich aufgebaut. Die DNS gehört zur Gruppe der Nukleinsäuren, die Polymere von Mononukleotiden sind. Jedes Mononukleotid enthält eine Base, eine Pentose

[4] 1822-1884

[5] Splicevarianten bedeutet, daß von einem Genort auf der DNS eines Organismus unterschiedliche Genprodukte erzeugt werden können. Splicevarianten enthalten gemeinsame Bereiche.

(Zucker) und eine oder mehrere Phosphatgruppen. DNS enthält die vier Basen *Adenin*, *Thymin*, *Guanin* und *Cytosin*, abgekürzt *A, T, G* und *C*. Watson und Crick erkannten, daß in der DNS *Adenin* und *Thymin* sowie *Guanin* und *Cytosin* immer im molaren Verhältnis von 1 vorkommen. Ihre Annahme, daß es zwischen diesen Basen zu Wasserstoffbrückenbindungen kommt, führte letztlich zur Erkenntnis, daß DNS als Doppelhelix vorliegen muß. Diese Doppelhelix besteht aus zwei spiralig antiparallel verlaufenden Polynukleotidketten, die, anschaulich gesprochen, die Geländer einer Wendeltreppe bilden; die Wasserstoffbrückenbindungen zwischen den Basen sind die Treppenstufen.

Bei prokaryoten Mikroorganismen ist die DNS ringförmig angeordnet und befindet sich im Cytoplasma. Die DNS höher organisierter (eukaryoter) Zellen ist im Zellkern organisiert. Je nach Organismus ist dieser DNS-Faden unterschiedlich lang. Beim Menschen erreicht er pro Zelle 2 Meter, verteilt auf 23 Stücke (*Chromosomen*). Damit entspricht die Länge der DNS eines Menschen ca. $2 \cdot 10^{11}$ km (bei 10^{14} Zellen).

2.1.3 Genexpression und Proteinbiosynthese

Die DNS enthält notwendige Informationen zur Synthese von Proteinen. Anhand dieses Bauplans wird ein Protein in der Zelle synthetisiert. Dies gilt für alle Organismen, ob Bakterien, Pflanzen oder Wirbeltiere, d.h. es hat keine Veränderung im Laufe der Evolution stattgefunden. Allerdings kann die Kodierung in verschiedenen Organismen unterschiedlich sein. Man kann daher nicht, ohne aufwendige Versuche in Laboratorien durchzuführen, davon ausgehen, daß Gene eines Organismus in einen anderen Organismus übertragen, für das gleiche Genprodukt kodieren.

Ist die Kodierung dagegen gleich, kann dies von großem Nutzen sein. In der modernen Tierzucht beispielsweise, in der nicht mehr auf dem Prinzip von Selektion und Auslese, sondern auf gentechnischem Wege eine zielgerichtete Züchtung durchgeführt wird, hat diese Tatsache zu einer Reihe von Anwendungen geführt. Ein wichtiges Einsatzgebiet seit Beginn der 80er Jahre ist die Erzeugung transgener Tiere oder von *knock-out* Mutanten, die als Modelle für menschliche Erkrankungen dienen können (z.B. “Krebsmaus”). Bei transgenen Tieren wurde das Genom durch gentechnische Methoden verändert. Sie sind unverzichtbares Hilfsmittel zur Erforschung entsprechender Krankheiten. Ein weiteres wichtiges Ziel ist es, pharmazeutisch wirksame Proteine in der Milch transgener Tiere (z.B. Rind, Schaf, Ziege) zu gewinnen.

In der DNS sind neben den Informationen über Primärsequenzen von Proteinen auch Verfahrensregeln über die Regulation von Genexpression und Proteinbiosynthese kodiert. Eine als *zentrales Dogma* der Molekularbiologie bezeichnete Annahme besagt, daß der gesamte Bauplan einer Zelle in der DNS niedergelegt

ist. Für die Biosynthese eines bestimmten Proteins wird jedoch nur der Teil der DNS als Vorlage herangezogen, der das Protein kodiert. Allerdings spielen nichtkodierende Regionen eine entscheidende Rolle, ob die kodierende Sequenz translatiert wird.

Bevor das Protein synthetisiert werden kann, muß zunächst die Vorlage auf der DNS, das *Gen*, in Form einer RNS abgeschrieben werden (*Transkription*). RNS, *Ribonukleinsäure*, ist im Gegensatz zur DNS einsträngig. Die Nukleotide enthalten die Basen *Adenin*, *Uracil*, *Guanin* und *Cytosin*, abgekürzt *A*, *U*, *G* und *C*. *Thymin* der DNS wird auf der RNS durch *Uracil* ersetzt.

Bei der Transkription kommt es unter Einfluß von Enzymen zur lokalen Denaturierung von DNS, d.h. die DNS wird in diesem begrenzten Bereich einsträngig. Dieser DNS-Teil dient als Matrize, von der aufgrund der komplementären Basenbindungen RNS erzeugt wird. Der transkribierte RNS-Faden, Boten-RNS oder messenger-RNS (kurz *mRNS*) genannt, gelangt an den Ort der Proteinbiosynthese (*Ribosomen*) und wird dort in die Aminosäuresequenz eines bestimmten Proteins übersetzt (*Translation*). Die denaturierte DNS wird mit Hilfe von Enzymen wieder zum Doppelstrang renaturiert.

Die vier Basen der DNS, *A*, *T*, *G* und *C* enthalten Informationen über die Abfolge der Aminosäuren im Protein. Da 20 verschiedene Aminosäuren für die Proteinbiosynthese zur Verfügung stehen, benötigt man mindestens drei Basen zur Kodierung ($4^3 > 20 > 4^2$). Diese Informationseinheit auf der DNS bestehend aus drei Aminosäuren nennt man *Basentriplett* oder *Codon*. Da nur 20 Amonisäuren kodiert werden müssen, existieren mehrere Codons für eine Reihe von Aminosäuren. So stehen z.B. die Codons *GAA*, *GAG*, *GAT*, *GAC*, *AAT* und *AAC* für die Aminosäure *Leucin*. Dieses Phänomen bezeichnet man als *Degeneration des genetischen Kodes*; er ist nicht eindeutig.

Im Gegensatz zu prokaryotischen Genen enthalten eukaryotische Gene nicht translatierte Sequenzen (*Introns*). Die kodierenden Bereiche eines Gens werden als *Exons* bezeichnet. Die bei der Transkription zunächst entstandene Vorlage enthält sowohl Exons als auch Introns. Anschließend werden aus dieser Vorlage die Introns entfernt und die Exons *gespleißt*.

Neben Genen enthält DNS weitere wichtige Informationen. Damit es zur Transkription kommen kann, muß sich ein Initiationskomplex bilden können. Dazu ist ein Bereich oberhalb des eigentlichen Gens auf der DNS von Bedeutung. Ebenfalls in dieser Region befinden sich *Enhancer*, die die Regulierbarkeit der Transkription spezifischer Gene vermitteln. Regulatorische Bereiche enthalten oft mehrere Bindungsstellen für verschiedene Transkriptionsfaktoren.

2.2 Genomsequenzierung: Vom Organismus zur genetischen Information

Proteine übernehmen vielfältige Aufgaben im Organismus (Struktur, Katalyse, Erkennung) und sind daher zentraler Gegenstand in der biomedizinischen Forschung. Zur Wahrnehmung der Aufgabe eines Proteins ist seine Raumstruktur entscheidend. Die Raumstrukturen von Proteinen sind sehr komplex, um die erforderlichen Funktionalitäten erfüllen zu können. Dagegen ist ihre Kodierung auf der DNS vergleichsweise einfach. Analog ist die Bestimmung der Struktur eines gegebenen Proteins mit den heute zur Verfügung stehenden Techniken relativ aufwendig. Im Gegensatz dazu wurden zur Analyse der DNS-Sequenz automatisierte und (relativ) kostengünstige Methoden entwickelt. Die Bioinformatik versucht, ausgehend vom Genom, Bereiche zu lokalisieren, die für Proteine kodieren. Diese Abschnitte werden sequenziert, analysiert und in Datenbanken verwaltet. Die dort enthaltenen Informationen werden der wissenschaftlichen Öffentlichkeit zur Verfügung gestellt.

Viele Krankheiten, wie z.B. Phenylketonurie,[6] sind auf Mutationen im Genom zurückzuführen, d.h. sie haben genetische Ursachen. Um diagnostische Methoden entwickeln und später auch Hoffnungen auf therapeutische Maßnahmen erfüllen zu können, müssen sowohl Gene als auch die Mechanismen ihrer Regulation im lebenden Organismus untersucht und verstanden werden. Eine Möglichkeit der Gentherapie in der Medizin besteht in der direkten Behandlung von Erkrankungen an ihrem Ursprungsort, den veränderten Genen. Bei der *Antisense*-Therapie wird die Synthese von Onkogenen verhindert, indem zur mRNS des transkribierten Onkogens (*sense*) eine komplementäre mRNS (*antisense*) erzeugt und in die Zelle eingeführt wird. Liegen diese beiden Formen vor, bilden sie einen Komplex; die Genexpression des Onkogens wird unterdrückt. Trotz ihrer Anfangserfolge[7] ist die Gentherapie bisher nicht aus ihrem experimentellen Stadium herausgekommen. Neben der Problematik, entscheiden zu müssen, ob ein Gen ersetzt oder ausgeschaltet werden soll, ist das Hauptproblem, die zusätzliche DNS sicher, effizient und dauerhaft an den gewünschten Ort zu transportieren. Dennoch werden mit diesem Therapiekonzept weiter große Hoffnungen verbunden.

Zelluläre Abläufe stellen komplizierte Netzwerke da, deren Modellierung eine weitere Herausforderung für die Informatik darstellt (z.B. Metabolische Netzwerke). Bisher erfolgen deren Darstellungen hauptsächlich als große statische Karten (siehe z.B. [Mic99]), die keine Möglichkeiten der Rechnerunterstützung (Vergleich, Simulation) erlauben. Erste Ansätze zur Dynamisierung sind derzeit im

[6] Autosomal-rezessiv erbliche Stoffwechselanomalie ([Zin86]).

[7] Die erste erfolgreiche Gentherapie wurde 1990 in den USA an einem vierjährigen Mädchen durchgeführt, das unter einem genetisch bedingten Immundefekt litt ([Har99], S. 45).

Entstehen (z.B. [FM99]).

Von besonderem Interesse ist die Bestimmung der Funktionen von Proteinen, für die kodierende Bereiche im Genom gefunden wurden. Dies ist i.d.R. nur mit aufwendigen und teueren Experimenten durchführbar. Auch der umgekehrte Weg ist von großem Interesse: die Lokalisation von Genen in Genomen, über deren Produkte genauere Informationen über ihre Funktionen in Zellen vorhanden sind. Der systematische Vergleich unterschiedlicher Organismen ist ein interessanter Aspekt in der Sequenzdatenanalyse. Man versucht Fragen zu beantworten wie „Haben verschiedene Organismen gleiche Gene oder ähnliche Proteine?“. Hat man in einem Organismus ein Protein gefunden, das in anderen auch vermutet wird, erleichtert dies evtl. die Suche. Durch solche intergenomischen Vergleiche wird der evolutionäre Hintergrund beleuchtet, so daß deduktive Ableitungen von Funktionen möglich werden.

Sequenzierung

Ziel der Sequenzierung ist die Entschlüsselung des Erbgutes, d.h. die Bestimmung der Basenpaarabfolge der zugrundeliegenden DNS. Für die Ermittlung der Primärstruktur werden in Laboratorien entweder bestimmte Teilbereiche einer Erbmasse oder das gesamte Genom systematisch sequenziert. Die dazu notwendigen Methoden wurden Mitte der 70er Jahre entwickelt und in den letzten Jahren weiter verbessert.

Zunächst mußte man sich auf kleine Bereiche eines Genoms beschränken, da die zur Verfügung stehenden Techniken Sequenzierungen kompletter Genome nicht erlaubten. Mittlerweile wurden diese Techniken so weit verbessert und vor allem automatisiert, daß selbst die systematische Sequenzierung des menschlichen Genoms nicht nur machbar wurde, sondern bereits durchgeführt wird. Dieses weltweit organisierte Humangenomprojekt (HGP), das von der *Human Genome Organisation (HUGO)* koordiniert wird, deren Mitglieder zahlreichen Ländern angehören, hat als Ziel, „durch die Kenntnis der genetischen Grundlagen die Entwicklung menschlicher Erkrankungen zu verstehen und die diagnostischen und therapeutischen Möglichkeiten bei individuellen Erkrankungen zu verbessern.“ ([Har99], S. 41). Die erste komplette genomische DNS-Sequenz eines menschlichen Chromosoms wurde im Dezember 1999 veröffentlicht ([DSR+99]).[8]

Wurde ein Genom oder Teil eines Genoms sequenziert, liegt die Primärsequenz vor. Aus der Sicht der Informatik ist dies eine endliche Zeichenreihe über dem Alphabet Γ der vier Nukleinsäuren *Adenin, Thymin, Guanin* und *Cytosin*: $\Gamma = \{A, T, G, C\}$.

[8] Die Sequenz des Chromosoms 22 hat eine Länge von 33,4 Mega-Basen. Sie enthält mindestens 545 Gene.

Wird die Nukleinsäuresequenz in die korrespondierende Aminosäuresequenz übersetzt, liegt das Alphabet Σ der 20 proteinogenen Aminosäuren vor.

2.3 Sequenzvergleich als Analysemethode

Eine Genomsequenz enthält die gesamte Information, die ein Organismus zum Leben benötigt. In diesem Abschnitt wird der Sequenzvergleich als eine Analysemethode in der molekularen Sequenzdatenanalyse vorgestellt. Die besondere Bedeutung der DNS- oder Proteinsequenz wird durch folgendes Axiom der Molekularbiologie deutlich ([Gus97], p. 212): „In biomolecular sequences (DNA, RNA, or amino acid sequences), high sequence similarity usually implies significant functional or structural similarity."

Allerdings ist zu beachten, daß nicht verwandte Sequenzen ähnliche Strukturen ausbilden können ([Coh95]).

Das Auffinden einer Teilzeichenreihe innerhalb einer endlichen Sequenz von Zeichen über einem endlichen Alphabet ist ein bekanntes Problem in der Informatik. Es wurden eine Vielzahl von Algorithmen entwickelt, die eine effiziente Mustersuche in großen Datenmengen, wie z.B. in Enzyklopädien, erlauben.[9]

Für die Beantwortung biologischer Fragestellungen haben die Datenstrukturen Suffix-Baum (erstmals als Positionsbaum beschrieben in [Wei73], siehe auch [Ukk95]), generalisierter Suffix-Baum und Suffix-Array ([MM93]) sowie Positionsbaumvarianten (z.B. [Heu96]) besondere Bedeutung erlangt ([Gus97], p. 156).

In der molekularen Sequenzdatenanalyse ist das exakte Finden einer Teilzeichenreihe zu restriktiv. Redundanz und Ähnlichkeit sind die zentralen Phänomene in der Biologie. Die große Mehrzahl der in der Natur noch vorhandenen Proteine sind das Ergebnis einer kontinuierlichen Folge genetischer Duplikationen mit anschließenden Modifikationen ([Doo90b]). Nicht nur Redundanz im Informationsgehalt verwandter Proteinsequenzen ist daher ein wesentliches Merkmal, sondern die Gesamtmenge der genomischen Informationen aller Organismen ist redundant ([Doo90a]). In diesem Anwendungsgebiet der Informatik hat das Finden ähnlicher Zeichenreihen besondere Bedeutung erlangt. *Ähnlich* bedeutet, daß bestimmte Unterschiede in zwei DNS- oder Proteinsequenzen zugelassen werden. Sind zwei Proteine in diesem Sinn ähnlich bzw. *homolog*, wird ein evolutionärer Zusammenhang angenommen.

Eine graphische Gegenüberstellung von Zeichenreihen, die im Rahmen der inexakten Mustersuche als Treffer gelten, wird als *Sequenzalignment* bezeichnet.[10] Henikoff und Henikoff [HH92] schreiben: „Among the most useful computer-based tools in modern biology are those that involve sequence alignments of prote-

[9] Für einen Überblick siehe z.B. [Gus97].

[10] Für eine formale Definition von Alignment siehe [Gus97], p. 216).

ins, since these alignments often provide important insights into gene and protein function."

Alignments können in Klassen eingeteilt werden. Beim *globalen Alignment* von Sequenzpaaren sind zwei Proteine aufgrund der Abstammung von einem gemeinsamen Vorfahren über ihre gesamte Sequenzlänge ähnlich. Das *lokale Alignment* betrachtet Segmente von Proteinen, die in Beziehung stehen. Werden mehr als zwei Sequenzen gegenübergestellt, spricht man von *multiplen Alignments*. Letztere werden z.B. bei Proteinfamilien, bei denen alle Elemente der Familie im Alignment aufgeführt werden, oder bei Datenbankanfragen angewendet.

Es gibt eine Vielzahl frei verfügbarer Programme, mit denen in öffentlichen biologischen Sequenzdatenbanken (Nukleinsäuredatenbanken, Proteinsequenzdatenbanken) zu einer gegebenen Sequenz nach ähnlichen Sequenzen gesucht werden kann: FASTA [Pea90], BLAST [AGM$^+$90], BLIMPS [WH92], CLUSTAL W [THG94]. Jedes der genannten Programme ist für bestimmte Teilbereiche der Sequenzdatenanalyse entwickelt worden.

Die Bedeutung der Sequenzdatenanalyse wird durch folgendes Zitat deutlich ([CELS95]): „Determing function for a sequence is a matter of tremendous complexity, requiring biological experiments of the highest order of creativity. Nevertheless, with only DNA sequence it is possible to execute a computer-based algorithm comparing the sequence to a database of previously characterized genes. In about 50% of the cases, such a mechanical comparison will indicate a sufficient degree of similarity to suggest a putative enzymatic or structural function that might be possessed by the unknown gene."

Der umfangreiche Sequenzvergleich ist nur durchführbar, wenn die entsprechenden Datensammlungen verfügbar sind. Im folgenden Abschnitt werden die wichtigsten Vertreter der molekularbiologischen Datenbanken vorgestellt.

2.4 Molekularbiologische Datensammlungen

Seit Bestimmung der ersten Proteinsequenzen durch Proteinsequenzierung in den 50er Jahren werden Sequenzdaten gesammelt. Die ersten Ausgaben einer solchen Datenbank an der *National Biomedical Research Foundation (NBRF)* erschienen in gedruckter Form als *ATLAS of Protein Sequences* ([Day78]).

In den letzten 20 Jahren hat sich die Entwicklung molekularbiologischer Datensammlungen drastisch verändert. Ein Artikel in der Zeitschrift *Science* beschreibt 1980 eine Nukleinsäuredatenbank mit 200 Einträgen und ca. 200.000 Basen, die der wissenschaftlichen Öffentlichkeit via Telefon kostenlos zur Verfügung gestellt wurde ([DSC$^+$80]). Ein Jahr später, 1981, erscheint in *Nature* eine Meldung mit dem Titel *Too many databanks?* (zitiert nach [FHLM98]). In der letz-

ten, auf molekularbiologische Datenbanken spezialisierten Ausgabe von *Nucleic Acids Research* (Volume 28, Issue 1, 2000), werden mehr als 100 unterschiedliche Datenbanken aufgelistet, die eine Vielzahl molekularbiologischer Forschungsbereiche abdecken. Die Nukleinsäuredatenbank *GenBank* enthält dabei mehr als 1 Milliarde Basen ([BBL+99]).

Nicht nur die Menge, Heterogenität und zunehmende Komplexität biologischer Daten beeinflussen Informationsrepräsentation, Speicherung, Anfragemöglichkeiten und Interpretationen der Daten. Auch die Benutzung dieser Ressourcen durch Wissenschaftler hat sich deutlich verändert. War in den 80er Jahren eine Datenbankabfrage auf Zentralrechnern nur von Experten durchführbar, die mit komplexen Anfrageschnittstellen umgehen konnten, ist mittlerweile die Nutzung freier Datenbanken vom Arbeitsplatzrechner aus tägliche Routine in der biologischen Forschung geworden. Die Entwicklung des *World-Wide Web* hat diese Revolution ermöglicht. Der Vorteil der Hypertext-basierten Informationsvermittlung liegt in der einfachen, intuitiven Bedienung. Selbst im Umgang mit Computern und dem Internet noch nicht erfahrenes Personal kann mit Hilfe geeigneter Browser-Software die über die Welt verteilten Dienste im Bereich Genomforschung effizient nutzen.

2.4.1 Nukleinsäuredatenbanken

Nukleinsäuredatenbanken sind Sammlungen von DNS- und RNS-Sequenzen. Ziele der Betreiber sind Vollständigkeit und Verteilung dieser für viele Bereiche der Forschung wichtigen Daten. Neue Einträge werden in der Regel direkt von einzelnen Wissenschaftlern[11] oder Genomsequenzierungskonsortien eingereicht, oder stammen aus Patentanträgen. An Bedeutung verliert die manuelle Extraktion neuer Sequenzen aus der wissenschaftlichen Literatur.

Diese Datensammlungen sind Archive von Nukleinsäuresequenzen. Änderungen, speziell Korrekturen oder Aktualisierungen der enthaltenen Informationen, liegen in den Händen derer, die Sequenzen in die Datenbank eingereicht haben. Es existiert keine Gruppe von Annotatoren, die den Datensatz pflegt (im Gegensatz zu Proteinsequenzdatenbanken, siehe Abschnitt 2.4.2).

Für diese Klasse von Datensammlungen existiert seit 1982 eine weltweite Kooperation zwischen der *EMBL Nucleotide Sequence Database*, seit 1992 am Europäischen Bioinformatik Institut (EBI) in Hinxton, UK [STLS99], *GenBank* am NCBI in Bethesda, USA [BBL+99], und der *DNS Database of Japan* in Mishima, Japan [SMGT99]. Jedes dieser Zentren ist für den festgelegten Einzugsbereich zuständig in dem Sinn, daß das Einreichen neuer Sequenzen ermöglicht und Tests auf syntaktische Korrektheit durchgeführt werden. Täglich werden Daten

[11] direct submissions

zwischen diesen Datenbankbetreibern ausgetauscht. Repräsentativ soll die für Europa zuständige *EMBL Nucleotide Sequence Database* vorgestellt werden (siehe auch [STLS99]).

EMBL Nucleotide Sequence Database

Die *EMBL Nucleotide Sequence Database* enthält mehr als drei Millionen Einträge mit mehr als zwei Milliarden Basenpaaren (Release 58, März 1999). Aufgrund systematischer Sequenzierungsprojekte verdoppelt sich die Menge der gespeicherten Sequenzen jährlich.

Die Datenbank ist in 17 disjunkte Kategorien unterteilt, wie z.B. Viren, Organellen, Pflanzen, Mensch, etc. und ist als Textdatei in definiertem Format erhältlich. Der Vertrieb als Textdatei wurde gewählt, um eine möglichst große Gruppe von Benutzern erreichen zu können und nicht an bestimmte Plattformen oder Datenbankmanagementsysteme gebunden zu sein. Dieses Datenformat wird von zahlreichen Analyseprogrammen unterstützt, die direkt auf Daten der Textrepräsentation biologischer Informationen zugreifen können.

Ein typischer Datenbankeintrag enthält eine kurze Beschreibung zu Katalogisierungszwecken, eine Beschreibung der Taxonomie[12] des zugrundeliegenden Organismus, Literaturinformationen, die Nukleinsäuresequenz sowie Tabellen, die kodierende und weitere biologisch interessante Bereiche der Nukleinsäuresequenz beschreiben. Proteine kodierender Bereiche werden eindeutige Protein Identifikationsnummern (PIDs) zugeordnet. PIDs können genutzt werden, um von externen Informationseinheiten auf entsprechende Einträge in Nukleinsäuredatenbanken zu verweisen.

Neue Sequenzen können auf elektronischem Weg (via *World-Wide Web*) beim Datenbankbetreiber eingereicht werden. Herausgeber wissenschaftlicher Journale fordern bereits die Einreichung der Sequenz an die Nukleinsäuredatenbank, bevor der auf die Sequenz verweisende Aufsatz publiziert werden darf. Im gedruckten Artikel wird auf den entsprechenden Eintrag verwiesen. Aufgrund dieses Verfahrens erscheinen neue Sequenzen einige Monate früher in digitalen Datenbanken als in Journalpublikationen.

Bei diesem Verfahren treten jedoch Inkonsistenzen auf: Sequenzen im gedruckten Aufsatz können sich von Sequenzen in Datenbanken unterscheiden. Datenbankeinträge können nachträglich korrigiert werden, gedruckte wissenschaftliche Publikationen nicht. Errata sind in schwerwiegenden Ausnahmefällen möglich, werden jedoch gewöhnlich nur in einer folgenden Ausgabe abgedruckt. Ein Verweis auf nachfolgende Korrekturen ist zum Zeitpunkt der Publikation des Ori-

[12] Taxonomie ist die Wissenschaft und Lehre von dem praktischen Vorgehen bei der Einordnung der Organismen in systematische Kategorien.

ginalartikels natürlich nicht möglich und daher im Nachhinein schwer zu entdecken.

2.4.2 Proteinsequenzdatenbanken

Proteinsequenzdatenbanken enthalten Sequenzen von Proteinen, von denen angenommen wird, daß sie in lebenden Zellen vorkommen und bestimmte Aufgaben im Organismus erfüllen. In den meisten Fällen fehlt jedoch der durch Experimente belegte Beweis. Proteinsequenzen besitzen korrespondierende Abschnitte auf der DNS und verweisen indirekt auf Einträge in Nukleinsäuredatenbanken.

Sequenzdatenbanken sind eine essentielle Informationsquelle, um nicht nur in wissenschaftlichen Projekten, sondern auch in der industriellen Forschung und Entwicklung Rückschlüsse von bereits existierendem Wissen auf aktuelle Forschungsobjekte zu ziehen. Z.B. werden bei der Suche nach neuen, bisher unbekannten Sequenzen in Organismen genetische Informationen zunächst nicht vollständig bestimmt. Anschließend erfolgen rechnergestützte Analysen, ob diese Bereiche bereits bekannt, d.h. als Einträge in einer der großen Basisdatensammlungen enthalten sind. Dadurch können Zeit und Kosten gespart werden. Man bedient sich der Tatsache, daß ähnliche Sequenzen ähnliche Proteine kodieren und damit ähnliche Funktionen in der Zelle besitzen. Wurde die Sequenz eines unbekannten Proteins bestimmt, können im globalen Datenraum biologisch ähnliche Sequenzen gesucht werden. Existieren solche Proteinsequenzen im Datenraum, ist es oftmals möglich, aufgrund der vorhandenen Annotationen der ähnlichen bekannten Proteine Rückschlüsse auf die Funktion des untersuchten Proteins zu ziehen.
Es gibt weltweit zwei konkurrierende Proteinsequenzdatenbanken:

- *PIR-International* [BGM+99]
- *SWISS-PROT* [BA99]

Im Gegensatz zu Nukleinsäuredatenbanken werden die biologischen Informationen in Proteinsequenzdatenbanken von spezialisierten wissenschaftlichen Gruppen gepflegt (*annotiert*). Proteinsequenzen werden von diesen Gruppenmitgliedern (*Annotatoren*) mit biologischen Zusatzinformationen, den *Annotationen*, versehen und ständig an den aktuellen Wissensstand angepaßt. Diese Pflege geschieht in einer konsistenten Art und Weise innerhalb der jeweiligen Annotationsgruppe der Proteinsequenzdatenbank. So ist z.B. die Verwendung eines kontrollierten Vokabulars in diesem Gebiet der molekularbiologischen Grundlagenforschung essentiell. Dadurch wird wissenschaftlichen Anwendern eine Möglichkeit gegeben, zwischen verschiedenen Proteinsequenzen vergleichen sowie generell

prozeßgesteuerte Weiterverarbeitungen durchführen zu können. Allerdings unterscheiden sich Annotationen zwischen unterschiedlichen Proteinsequenzdatenbanken sowohl inhaltlich, als auch im Format, in dem die Annotationen bereitgestellt werden. Dadurch wird ein direkter Vergleich zwischen Proteinsequenzen aus verschiedenen Proteinsequenzdatenbanken erschwert.

In dieser Arbeit sollte die Annotationsgruppe der Proteinsequenzdatenbank *PIR-International* durch ein Groupwaresystem unterstützt werden. Im folgenden wird daher diese Datenbank vorgestellt.

PIR-International

Die Datenbank *PIR-International*, eine Abkürzung für *Protein Information Resource*, existiert seit 1984. Gegründet wurde diese Initiative von der *National Biomedical Research Foundation (NBRF)* in den USA. Sie geht zurück auf Sammlungen, die bereits 20 Jahre zuvor von Margaret O. Dayhoff initiiert, und als *Atlas of Protein Sequence and Structure* in gedruckter Form publiziert wurden.

Seit 1988 besteht *PIR-International*, eine Assoziation von Zentren, die sich mit der Sammlung von Proteinsequenzen beschäftigen. Neben der *Japan International Protein Information Database (JIPID)* in Japan, ist das *Münchener Informationszentrum für Proteinsequenzen (MIPS)* ein wichtiger Partner. Das in dieser Arbeit entwickelte Groupwaresystem soll die Gruppe bei MIPS unterstützen, die den europäischen Beitrag für *PIR-International* leistet.

Folgendes Zitat stellt die Aufgabe von *PIR-International* dar ([BGM+99]):

„(i) to create and maintain the Protein Sequence Database as a comprehensive, non-redundant, well verified collection, organized according to biological principles, including structural, functional and evolutionary relationships; (ii) to provide a research tool that supports the study of protein sequences, their structural and functional properties, and their biological origins; (iii) to freely distribute the database to the public by the most accessible means including the PIR Web site [...] and CD-ROM; and (iv) to collaborate with other databases in organizing and coordinating the presentation of biomolecular structural information."

Besonders hervorzuheben ist hier die Prämisse, die Organisation der Daten gemäß der zugrundeliegenden biologischen Prinzipien durchzuführen. Dies ist bei Modellierung durch den Informatiker zu beachten.

Neben der Qualität der in der Datensammlung enthaltenen Daten hat die Bereitstellung für die wissenschaftliche Öffentlichkeit eine besondere Bedeutung. Wurden die Daten zunächst ausschließlich auf magnetischen Bändern angeboten, gab es seit 1992 regelmäßige Versionen via `ftp` und auf CD-ROM. Mit der Etablierung des *WWW*, besonders im wissenschaftlichen Bereich der biologischen Forschung, wurde interaktiver Zugriff nicht nur ermöglicht, sondern stellt derzeit den wichtigsten Zugriffsmechanismus auf aktuelle biologische Informationen dar.

Auf Massenmedien, wie etwa CD-ROM, sind Informationen zu einem bestimmten Zeitpunkt, dem Releasedatum, enthalten (*snapshots*). Die Datensammlungen werden jedoch ständig um neue Einträge erweitert. Diese Art der Distribution ist aufgrund der zunehmenden Vernetzung und der damit verbundenen Möglichkeit, direkt auf neueste Daten zugreifen zu können, scheinbar obsolet geworden. Aus Sicherheitsgründen ist diese Form der Datenverteilung jedoch gerade für kommerzielle Nutzer interessant. Im Bereich der industriellen Forschung wird aus patentrechtlichen Gründen sehr viel Wert auf Sicherheit der eigenen Daten gelegt. Die Übertragung firmeneigener Sequenzen über das öffentliche Internet an einen Rechner eines akademischen Instituts zur Durchführung von Analysen ist meistens nicht praktikabel. Alternativ können sichere Kommunikationskanäle angeboten werden (z.B. dedizierte Telefonleitungen oder *secure http* (zu Sicherheitsaspekten siehe z.B. [CZ95])).

Neben der Vollständigkeit des Datenbestandes zeichnet eine Datensammlung besonders die Qualität der Annotationen aus. Um die Masse der schon enthaltenen Daten zu strukturieren, werden Proteine in Familien und Superfamilien eingeteilt. Dadurch erfolgt eine Strukturierung der Proteinsequenzen gemäß ihrem evolutionären Hintergrund. Alle Sequenzen von *PIR-International* sind in Proteinfamilien und Proteinsuperfamilien klassifiziert. Proteinfamilien und Proteinsuperfamilien bilden ein disjunktes hierarchisches System. Ein Protein ist genau in einer Proteinfamilie, eine Proteinfamilie in genau einer Proteinsuperfamilie enthalten.
Folgende Merkmale zeichnen *PIR-International* aus:

- es wird ein definiertes Vokabular für die Beschreibung biologisch relevanter Informationen eingesetzt;
- es werden Querverweise zu anderen Datenbanken, z.B. zu Nukleinsäuredatenbanken, eingetragen, falls eine Proteinsequenz mit einer veröffentlichten Nukleinsäuresequenz assoziiert werden kann; diese Querverweise müssen gewartet werden;
- neue Daten werden, so schnell wie es die interne Prozessierung erlaubt, veröffentlicht.
- in den Datensätzen wird nicht nur auf relevante Literaturstellen verwiesen, sondern sie sind mit Titel und Autorenzeilen enthalten;
- Querverweise auf öffentliche Literaturdienste, wie z.B. MEDLINE, über die auch Zusammenfassungen der wissenschaftlichen Artikel (*abstracts*) erhältlich sind, sind enthalten;

2.4.3 Spezialisierte Datenbanken

Neben den beiden Hauptvertretern der biologischen Datensammlungen, den Nukleinsäure- und den Proteinsequenzdatenbanken, existieren noch eine Vielzahl spezialisierter Sammlungen. Im folgenden wird eine kleine Auswahl solcher Datensammlungen präsentiert:

- *MYGD:* Die MIPS Hefe-Genomdatenbank *(Saccharomyces cerevisiae)* realisiert eine Wissensbasis, in der die genomische Struktur der vollständig sequenzierten Bäckerhefe enthalten ist.[13]
- *MATDB:* Die MIPS *Arabidopsis thaliana* Datenbank enthält die Sequenzen der Chromosomen 3, 4 und 5, die innerhalb des europäischen Sequenzierungsprojektes bestimmt wurden.[14]
- *EcoCyc:* Eine Enzyklopädie von *Escherichia coli* Genen und Metabolismen ([KRP+99])
- *KEGG:* Die *Kyoto Encyclopedia of Genes and Genomes* ist eine Wissensbasis, die systematische Analysen von Genfunktionen enthält. Funktionelle Informationen höherer Ordnung sind in einer Stoffwechseldatenbank repräsentiert, die graphisch dargestellt werden können ([KG00])
- *MGD:* Datenbank, die genetische und genomische Informationen über die Labormaus enthält ([BRDE99]); *MTB:* Daten, die Maustumoren betreffen ([BKE99])
- MITOP: Mitochondrien-bezogene Daten über Proteine, Gene und Krankheiten ([SZH+99])

Es existieren in diesem Bereich der Gen- und Genomanalyse weitere Forschungsbereiche, auf die hier nicht eingegangen werden kann (z.B. Sekundärstrukturvorhersage, Erstellen von Genmodellen).

Mit diesem Kapitel wurde der biologische Hintergrund sowie das Anwendungsgebiet der molekularbiologischen Sequenzdatenanalyse vorgestellt. Im folgenden Kapitel werden Anforderungen an ein CSCW-System zur Unterstützung in diesem Anwendungsgebiet herausgearbeitet.

[13] http://www.mips.biochem.mpg.de/proj/yeast/
[14] http://www.mips.biochem.mpg.de/proj/thal/

Kapitel 3

CSCW in der Biologie: Anforderungen und existierende Ansätze

In diesem Kapitel werden, nach einer kurzen Einführung in das Forschungsgebiet Rechnergestützte Gruppenarbeit, *Gruppen in der Gen- und Genomanalyse definiert und allgemeine Anforderungen dieser Gruppen an ein CSCW-System formuliert. Anhand exemplarischer Situationen dieser Gruppen werden konkrete Anforderungen an ein Groupwaresystem für den alltäglichen Einsatz in der molekularbiologischen Sequenzdatenanalyse abgeleitet. Bereits existierende Ansätze werden dargestellt und vor dem Hintergrund des aufgestellten Anforderungskataloges unter dem Gesichtspunkt CSCW erstmals diskutiert. Eine zusammenfassende Diskussion schließt dieses Kapitel ab.*

3.1 Das Forschungsgebiet *Rechnergestützte Gruppenarbeit*

3.1.1 *Computer Supported Cooperative Work* - CSCW

In der Informatik ist die Untersuchung rechnergestützter Gruppenarbeit eine vergleichsweise junge Disziplin. Im allgemeinen wird unter dem Begriff *Computer Supported Cooperative Work* das Forschungsgebiet als solches verstanden, während *Groupware* bzw. *CSCW-Applikation* die in diesem Bereich entwickelte Software bezeichnet.

Das Forschungsgebiet CSCW befaßt sich mit Fragestellungen der Rechnerunterstützung von Gruppen, Entwicklungen von Software sowie deren Einsatz in Bereichen, in denen Gruppen in ihrer alltäglichen Arbeit durch Rechner un-

terstützt werden sollen. Voraussetzung für den Einsatz eines CSCW-Systems ist neben einer Gruppe von Personen vor allem ein gemeinsames Gruppenziel.

CSCW als Forschungsgebiet beschränkt sich nicht nur auf technische Bereiche der Informatik (wie z.B. Kommunikationstechnologie und Softwaretechnik), sondern ist stark interdisziplinär ausgerichtet. Psychologische und soziologische Aspekte werden dabei ebenso betrachtet wie betriebswirtschaftliche Faktoren im unternehmensweiten Einsatz. Besonderheiten des Anwendungsgebietes, in dem ein Groupwaresystem zum Einsatz kommen soll, müssen beim Entwurf ebenfalls beachtet werden.

Auf sprachlicher Ebene von CSCW gibt es eine Diskussion innerhalb der wissenschaftlichen Öffentlichkeit. In Anlehnung an Borghoff und Schlichter ([BS98]) wird in dieser Arbeit CSCW wie folgt verstanden:

Grundsätzlich kann der Begriff CSCW in *Computer Supported* und in *Cooperative Work* unterteilt werden. Bevor eine Unterstützung durch Rechner realisiert werden kann, muß kooperative Arbeit innerhalb einer Gruppe vorliegen.

Kooperative Arbeit liegt vor, wenn Mitglieder einer Gruppe, die aus mindestens zwei Gruppenmitgliedern besteht, ein gemeinsames Ziel, das *Gruppenziel*, verfolgen. Die Gruppenmitglieder können innerhalb der Gruppe für gleiche oder unterschiedliche Aufgabengebiete zuständig sein und sich auf gleichen oder verschiedenen Hierarchiestufen innerhalb der internen Organisation befinden. Ein gemeinsames Gruppenziel ist jedoch notwendige Voraussetzung. Um dieses Ziel erreichen zu können, ist Informationsaustausch durch Kommunikation zwischen Mitgliedern erforderlich.

Diese Kommunikation kann nach Intensität kategorisiert werden ([BS98], S. 111f): den geringsten Kommunikationsgrad hat das Informieren, bei der Information nur in eine Richtung fließt. Werden Informationen und Aktionen koordiniert, liegt als nächst höherer Kommunikationsgrad das Koordinieren vor. Arbeiten Sender und Empfänger auf ein Gruppenziel hin, befindet man sich auf der Stufe des Kollaborierens. Die höchste Stufe des Kommunikationsgrades liegt schließlich im Kooperieren. Darunter wird in der Grundbedeutung jede Art der Zusammenarbeit verstanden, die zwischen einzelnen Mitgliedern, Gruppen und Organisationen stattfindet. Im Bereich CSCW wird diese Kooperation koordinierend unterstützt, etwa durch gemeinsame Ziele oder Pläne sowie der Bearbeitung gemeinsamer Datenbestände. Die Unterstützung von *face-to-face*-Sitzungen durch das CSCW-System hat zentrale Bedeutung.

Mit wachsendem Kommunikationsgrad steigt die Möglichkeit der Unterstützung durch Rechner. Gemäß dieser Kategorisierung liegt in dem hier beschriebenen Fall der Kommunikationsgrad der Kooperation vor. Es wird nicht nur auf ein gemeinsames Gruppenziel hingearbeitet, sondern wichtige Entscheidungen werden durch einen Gruppenkonsens gefällt, der von allen Gruppenmitgliedern mitgetragen wird. Das individuelle Ziel ist dem Gruppenziel untergeordnet. Es be-

steht auch in der Kooperation noch genügend Freiraum für individuelle Ausführungen gruppeninterner Aufgaben. Speziell für Gruppen im wissenschaftlichen Anwendungskontext ist diese Forderung notwendig. Das spezialisierte Fachwissen eines Gruppenmitglieds soll gewinnbringend in den gemeinsamen Informationsraum einfließen können und nicht durch eine Kooperationsvorschrift beschnitten werden. Diese Freiräume müssen geeignet modelliert werden, z.B. durch Einführen einer Untergruppe in der Gruppenhierarchie.

Kooperative Arbeit (*Cooperative Work*) soll durch Rechner unterstützt werden (*Computer Supported*). Diese Unterstützung findet hauptsächlich auf den Ebenen der Kommunikation und der Koordination statt. Durch rechnergestützte Kommunikation soll es den Mitgliedern einer Gruppe ermöglicht bzw. erleichtert werden, das gemeinsame Gruppenziel zu erreichen. Da die primäre Interaktion in einem verteilten asynchronen CSCW-System zunächst mit dem Rechner erfolgt, muß das Wissen des Gruppenmitglieds um seine Existenz in einer Gruppe und der Handlungen ihrer Mitglieder durch das CSCW-System vermittelt werden (*group awareness*, [BS98]). Dieses psychologische Existenzbewußtsein muß durch Software geeignet erzeugt und gleichzeitiges Handeln von Gruppenmitgliedern unter Sicherstellung der Konsistenz unterstützt werden. Es ist das Miteinander entscheidend, auch wenn jedes Mitglied zunächst scheinbar alleine an einem Rechner am Arbeitsplatz, von zu Hause aus (Telearbeit) oder an mobilen Computern (Laptops, Notebooks) unterwegs an seinem Aufgabengebiet arbeitet. Gerade die zunehmende Bedeutung räumlicher Verteilung von Gruppenmitgliedern, wie z.B. aufgrund von Telearbeit ([RMS+98]), oder die zunehmend wegfallenden starren Arbeitszeitregelungen erfordern eine entsprechende kommunikative Unterstützung durch Software, um geeignete Zusammenarbeit zu erreichen.

Generell kann Unterstützung im Bereich CSCW unterschiedlich verstanden werden (vgl. [BS98], S. 113): sie kann im Finden einer besten Lösung aus einer Vielzahl kontroverser Meinungen bestehen oder im Generieren einer geeigneten Umgebung für kreative Vorschläge (z.B. *brainstorming*). Unterstützung kann aber auch das Strukturieren von Informationen bedeuten und die Beachtung von Regeln erzwingen.

Zur Überwindung der Grenze von Raum und Zeit müssen in erster Linie das anfallende Kommunikationsproblem gelöst ([RMS+98]), sowie trotz räumlicher Trennung und asynchroner Arbeit ein Gruppenbewußtsein erzeugt werden.

3.1.2 CSCW und Groupware

Bezeichnet *CSCW* das Forschungsgebiet, so ist mit *Groupware* oder *CSCW-Applikation* ein konkretes System gemeint, das für einen bestimmten Anwen-

dungsbereich realisiert wurde.[1]

Der Einsatz von Rechnern im Berufsalltag erleichtert viele Aufgaben. Neben Standardanwendungen, wie Unterstützung in der Erstellung eines Dokuments oder die Verwaltung von Daten in Datenbanken, spielt die rechnergestützte Kommunikation aufgrund zunehmender Vernetzung von Arbeitsplatzrechnern eine immer wichtigere Rolle. Nach [TSMB95] sind unter dem Begriff Groupware nicht Applikationen einzuordnen, mit denen mehrere Benutzer gleichzeitig, aber voneinander unabhängig arbeiten können. Diese Trennung zwischen Groupware-Applikationen und Mehrbenutzer-Applikationen[2] ist oft schwer durchzuführen. Es hängt davon ab, wie eng die Kooperation zwischen Anwendern definiert wird.[3]

Die bekannteste und am weitesten verbreitete Unterstützung einer asynchronen Kommunikationsform ist das elektronische Postsystem (*Email*). Die Vorzüge liegen nicht nur in der leichten Komposition, der schnellen Übertragung, speziell über Länder- und Kontinentgrenzen hinweg, und Benachrichtigung sowie der freien Entscheidung über den Bearbeitungszeitpunkt, etwa im Gegensatz zu einem Telefonanruf, sondern auch in den geringeren formalen Anforderungen im Geschäftsbereich. Das gleichzeitige Versenden an eine Menge von Empfängern (Viele-zu-viele-Kommunikation) und das Einstreuen von Erwiderungen in die Originalnachricht sind weitere Gründe für die hohe Akzeptanz dieser Kommunikationsform.

Ein weiteres Beispiel für die asynchrone Kommunikationsunterstützung ist das kollaborative Erstellen eines Dokuments. Oftmals sind Autoren nicht nur räumlich verteilt, sondern befinden sich auch in unterschiedlichen Zeitzonen. Das gleichzeitige Bearbeiten eines Dokuments muß durch ein System unterstützt werden, um u.a. Konsistenz gewährleisten zu können (siehe z.B. [Koc96]).

Aufgrund der bisher aufgeführten Aspekte, wie Kommunikationsunterstützung und Erzeugung von *group awareness*, wird deutlich, daß das Forschungsgebiet CSCW neben der Hauptdisziplin Informatik weitere Disziplinen beachten muß, um die gestellten Anforderungen erfüllen zu können.

[1] Definition *Groupware* nach [GR91] „Groupware are computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment."

[2] „Eine Mehrbenutzer-Applikation ist ein Computersystem, mit dem mehrere Benutzer gleichzeitig, aber voneinander unabhängig arbeiten können" ([TSMB95], S. 22).

[3] Karl Marx beispielsweise definiert Kooperation in *Das Kapital* folgendermaßen: „Die Form der Arbeit vieler, die in demselben Produktionsprozeß oder in verschiedenen, aber zusammenhängenden Produktionsprozessen, planmäßig neben- und miteinander arbeiten, heißt *Kooperation*." [Mar57], S. 215.

3.1.3 Interdisziplinarität von CSCW

Soll kooperative Arbeit durch Rechner unterstützt werden, sind neben technischen eine Vielzahl weiterer Aspekte zu beachten. In den Anfängen von CSCW wurde zu wenig Augenmerk auf die Erforschung der Zusammenarbeit von Menschen gelegt. So schreibt Henninger beispielsweise:

„The main problem (...) was a lack of analysis of work environments to motivate the need for systems. In the systems work, there seems to be a focus on the technology for the sake of technology.“ [Hen91].

Bei der Entwicklung eines CSCW-Systems müssen neben der Komponente Mensch als weitere Komponenten die Organisation, die Technik sowie die Aufgabenausführung der unterstützten Gruppen analysiert werden. Neben der Untersuchung der Zusammenarbeit von Menschen, kann vor allem die empirische Sozialforschung wichtige Beiträge bei der Untersuchung der Akzeptanz von CSCW-Systemen liefern. Ein Groupwaresystem muß sich nach Einführung der Kritik der Gruppenmitglieder stellen. Eine Untersuchung über das hier entwickelte Groupwaresystem in der Gen- und Genomanalyse wurde daher unter Verwendung einer ausgewählten Methode der empirischen Sozialforschung durchgeführt (siehe Kapitel 6.6.3).

Die große Herausforderung bei Entwurf und Realisierung eines Groupwaresystems liegt in der harmonischen Integration der technischen sowie der sozialen Welt ([Nö98]). Vor dem Entwurf eines neuen Systems müssen Untersuchungen herangezogen werden, die sich mit den Gründen sowohl der Akzeptanz, als auch der Nicht-Aktzeptanz von CSCW-Systemen beschäftigen.

Nach Grudin ([Gru93]) existieren eine Reihe von Faktoren, weshalb der Einsatz von Groupwaresystemen fehlschlagen kann: Die Einführung eines CSCW-Systems führt oftmals zu Mehrarbeit bei Gruppenmitgliedern, die keinen unmittelbaren Nutzen durch den Einsatz sehen. Werden soziale Tabus oder bereits existierende organisatorischen Strukturen verletzt, kann Demotivation die Folge sein. Gruppenaktivität zeichnet sich oft durch die Behandlung von Sonderfällen und durch Improvisation aus. Beide Aspekte sind schlecht modellierbar. Wurden Erfahrungen gesammelt und fruchtbare Analysen zur Verbesserung durchgeführt, stellt die Komplexität des Softwaresystems oftmals eine unüberwindliche Barriere dar, die eine Verbesserung des Systems unmöglich macht.

Neben diesen eher sozialen und technischen Faktoren spielen betriebswirtschaftliche Aspekte, wie z.B. Kosten/Nutzen-Fragen für Unternehmen eine wichtige Rolle bei der Entscheidung, ob ein CSCW-System eingesetzt werden soll.[4] Eine umfassende Analyse muß daher zu Beginn der Entwurfsphase durchgeführt werden.

[4]Diese Fragen werden von einem Teilgebiet der Informatik, der Wirtschaftsinformatik, untersucht und daher in dieser Arbeit nicht behandelt (siehe z.B. [Wag95]).

Trotz der vielschichtigten Interpretationen des Begriffs *CSCW* ist der Aspekt, daß manche Ziele ohne Einsatz eines geeigneten Groupwaresystems mit vertretbarem Aufwand nicht erreicht werden könnten, hervorzuheben. Dazu zählt das hier vorliegende Anwendungsgebiet. So ist beispielsweise die Annotation aller weltweit bekannten und öffentlich zugänglichen Proteinsequenzen durch eine kleine Gruppe ohne geeignete kommunikative Unterstützung nicht praktikabel durchführbar. Durch den Einsatz von CSCW-Systemen können daher nicht nur bestehende Gruppenziele besser unterstützt, sondern auch das Erreichen neuer Ziele ermöglicht werden.

3.1.4 Klassifizierung von CSCW-Systemen

Seit den Anfängen des Forschungsgebietes CSCW wird versucht, die große Vielfalt von CSCW-Systemen durch Klassifizierungen übersichtlicher zu gestalten. Erschwert wird dieser Ansatz dadurch, daß CSCW-Anwendungen viele, oft unterschiedliche Bereiche der Gruppenarbeit abdecken.

In den bekanntesten Klassifikationsschemata wird nach den Zeitbedingungen der Interaktionen und der Verteilung der Gruppenmitglieder unterschieden. Bei der *synchronen Gruppenarbeit* erfolgt eine Kommunikation zur gleichen Zeit, bei der *asynchronen Gruppenarbeit* ist sie zeitverschieden.

Gruppenmitglieder können am gleichen Ort oder an verschiedenen Orten arbeiten. Diese Aufteilung ist nicht immer eindeutig. So kann gleicher Ort bedeuten, daß die Gruppenmitglieder im gleichen Zimmer sitzen, oder auf dem gleichen Stockwerk, oder auf unterschiedlichen Stockwerken des gleichen Gebäudes. Gruppen können über mehrere Gebäude innerhalb einer Stadt oder über mehrere Städte innerhalb eines Landes verteilt sein. Es ist bei der Kommunikationunterstützung zu beachten, daß Gruppenmitglieder evtl. in unterschiedlichen Zeitzonen arbeiten.

Der stark expandierende Markt der Online-Anbieter nutzt diese technischen Möglichkeiten, um eine bessere Verfügbarkeit für den Kunden und damit Wettbewerbsvorteile zu realisieren. Es existieren Ableger von Online-Geschäften in verschiedenen Zeitzonen, um einen 24-Stunden Service via Internet anbieten zu können. Für die Kunden bleibt dabei transparent, in welcher Zeitzone sich sein Kommunikationspartner befindet.

Je nach Art der Verteilung muß die Kommunikation zwischen Gruppenmitgliedern adäquat unterstützt werden. Dazu sind ein geeignetes Kommunikationsmedium sowie eine geeignete Kommunikationsform notwendig.

Ein weiterer Klassifizierungsaspekt ist die Interaktionsart. Bei der *expliziten Kommunikation* erfolgt ein aktiver Informationsaustausch, etwa innerhalb eines Gespräches oder über elektronische Post. *Implizite Kommunikation* liegt vor, falls

sie über gemeinsame Informationsobjekte durchgeführt wird, die beispielsweise in einer Datenbank abgelegt sind ([TSMB95] S. 24).

Bekannte Klassifikationsschemata sind die Raum-Zeit-Matrix nach Johansen [Joh88] oder eine funktionelle Klassifikation nach Ellis et al. ([GR91]).

Nach Teufel et al. ([TSMB95]) lassen sich überschneidende Systemklassen extrahieren. Die Systemklasse *Kommunikation* ermöglicht den expliziten Informationsaustausch zwischen verschiedenen Kommunikationspartnern. Informationen, die längere Zeit in geeigneter Form und mit Hilfe geeigneter Zugriffsmechanismen gespeichert sind, und einer Gruppe zur Verfügung gestellt werden, werden durch die Systemklasse *gemeinsame Informationsräume* beschrieben. *Workflow Management* umfaßt alle Aufgaben, die bei der Ausführung und Steuerung einer endlichen Folge von Aktivitäten entstehen. *Workgroup Computing-Systeme* schließlich unterstützen Gruppen, deren Aufgaben einen geringen Strukturierungsgrad oder eine geringe Wiederholungsfrequenz aufweisen.

Darüberhinaus kann nach der Größe der Gruppe und ihrer Zusammensetzung klassifiziert werden. Es ist zu untersuchen, inwieweit sich Gruppenmitglieder hinsichtlich ihres Ausbildungsniveaus und ihrer hierarchischen Position innerhalb der Gruppe unterscheiden. Es wird unterschieden, ob die Kommunikation innerhalb fester Regeln ablaufen soll oder informell durchgeführt werden kann.

Neben diesen quantitativen und sozialen Klassifikationen gibt es schließlich noch organisatorische Einteilungen, die sich beispielsweise mit der Durchführung von Sitzungen beschäftigen.

Aus technischer Sicht muß aufgrund der zunehmenden Vernetzung von einer *verteilten rechnergestützten Gruppenarbeit* gesprochen werden. Die in einem Netzwerk zur Verfügung stehenden Ressourcen sollen neben der verbesserten Datenverarbeitung durch Verteilung auch für rechnergestützte Kooperationen genutzt werden. CSCW erleichtert den Informationsaustausch von verteilten Personen unter Einsatz von Kommunikationsnetzen und -protokollen. In einem solchen System existieren eine Vielzahl von Komponenten, die auf unabhängigen Rechnern ablaufen und miteinander über ein Netzwerk kommunizieren. Als Beispiele seien hier nur der entfernte Datenbankzugriff oder Koordinierungsaktivitäten innerhalb der Gruppe genannt.

Auf dieser technischen Ebene der Kommunikation sind daher Techniken, wie der entfernte Prozeduraufruf (*Remote Procedure Call*), von zentraler Bedeutung (siehe z.B. [Sch92a], [Sch92b]). Es gibt seit einigen Jahren Bestrebungen, industrielle Standards zu etablieren, um eine Kommunikation zwischen Produkten unterschiedlicher Hersteller realisieren zu können. Ein Beispiel ist die *Common Object Request Broker Architecture (CORBA)* der *Object Management Group (OMG)*.

3.2 Gruppen in der Gen- und Genomanalyse

Der Einsatz eines Groupwaresystems hat zum Ziel, aufgrund der rechnerbasierten Unterstützung alltäglicher Arbeiten, insbesondere der Unterstützung der Kommunikation zwischen den Gruppenmitgliedern, das gemeinsame Gruppenziel zeitlich früher und qualitativ besser zu erreichen als ohne. Bevor ein solches System entworfen und realisiert werden kann, müssen die Aufgaben und die Gruppen, für die das System entwickelt werden soll, analysiert und Anforderungen an ein Groupwaresystem abgeleitet werden. Dazu muß sich der Informatiker mit dem Anwendungsgebiet vertraut machen, um eine gemeinsame sprachliche und inhaltliche Ebene mit Vertretern des Anwendungsgebiets zu erreichen.

Das hier beschriebene Groupwaresystem ist im Bereich Gen- und Genomanalyse angesiedelt. Aspekte von CSCW wurden in der Bioinformatik bisher jedoch nicht betrachtet. Mit dieser Arbeit soll ein erster Beitrag zu einem Wissenstransfer in ein für CSCW neues Anwendungsgebiet geleistet werden.

Der verbleibende Teil dieses Kapitels beschäftigt sich mit Gruppen, für die das System entworfen und realisiert werden sollte. Es werden deren Aufgaben beschrieben und allgemeine Forderungen an ein System zur Unterstützung herausgearbeitet, die generell die Anforderungen an eine in diesem Anwendungsbereich tätigen Gruppe beschreiben. Aus der Analyse von Anwendungsszenarien werden konkrete Anforderungen an das Groupwaresystem abgeleitet und ein Anforderungskatalog erstellt. Die in der Bioinformatik existierenden Ansätze werden anhand des erarbeiteten Anforderungskataloges unter dem Gesichtspunkt CSCW diskutiert.

3.2.1 Definitionen

Die Begriffe *Information* und *Informationsraum* haben zentrale Bedeutung. In diesem Abschnitt werden sie definiert. Generell ist das Ziel des CSCW-Systems die Unterstützung in der Generierung qualitativ hochwertiger biologischer Informationen aus Rohdaten, deren Verwaltung und Pflege sowie der Informationsaustausch zwischen Mitgliedern gleicher oder verschiedener Gruppen. Informationen werden in Informationsräumen verwaltet (nach [Nö98]):

Definition 1 (Biologische Information)
Strukturiertes Wissen über biologische Sachverhalte oder Vorgänge.

Definition 2 ((Biologischer) Informationsraum)
Ein (biologischer) Informationsraum ist eine Menge von inhaltlich vernetzten (biologischen) Informationen und damit in Zusammenhang stehenden Informationsdiensten, die für den Informationsaustausch nutzbar sind.

Definition 3 (Privater (biologischer) Informationsraum)
Der private (biologische) Informationsraum ist der private Informationsbereich eines Gruppenmitglieds, für den es eigenständig verantwortlich ist in Bezug auf Verwaltung und Bereitstellung der (biologischen) Informationen.

Definition 4 ((Biologischer) Gruppeninformationsraum)
Die in Gruppen arbeitenden Mitglieder haben je nach Gruppenzugehörigkeit Zugriff auf den (biologischen) Gruppeninformationsraum. Sie können enthaltene (biologische) Informationen nutzen und modifizieren sowie neue (biologische) Informationen importieren.

Definition 5 (Organisationsweiter (biologischer) Informationsraum)
Der organisationsweite (biologische) Informationsraum ist die Vereinigung aller (biologischen) Gruppen-Informationsräume einer Organisation.

Definition 6 (Globaler (biologischer) Informationsraum)
Der globale (biologische) Informationsraum ist die Vereinigung aller organisationsweiten (biologischen) Informationsräume.

In dieser Arbeit wird als globaler Informationsraum die Vereinigung aller organisationsweiten Informationsräume verstanden, die im Bereich der molekularbiologischen Sequenzdatenanalyse von Bedeutung sind. Dazu zählen beispielsweise die Nukleinsäure- und Proteisequenzdatenbanken, aber auch Literatursammlungen wie MEDLINE. Auf entfernte Gruppen-Informationsräume, d.h. auf Informationsräume einer anderen Gruppe, der ein Mitglied nicht angehört, ist lediglich lesender Zugriff erlaubt, es sei denn, Modifikationen sind explizit zugelassen. Der Zugriff auf private Informationsräume kann vom verantwortlichen Gruppenmitglied für weitere Mitglieder der eigenen oder fremder Gruppen verweigert werden. Konzeptionell haben alle Mitglieder der entsprechenden *community*[5] Zugriff auf den globalen Informationsraum der jeweiligen *community*, unter Berücksichtigung der jeweiligen Zugriffsrechte.

3.2.2 Wartung einer Proteinsequenzdatenbank

Wie in Abschnitt 2.4.2 beschrieben, enthält die Proteinsequenzdatenbank *PIR-International* annotierte Proteinsequenzen. Diese biologischen Informationen werden von einer wissenschaftlichen Gruppe gepflegt und in einem Gruppeninformationsraum verwaltet. Um die große Menge der bisher in den Datenbanken enthaltenen Proteinsequenzen (ca. 150.000) zu strukturieren, erfolgt eine Klassifikation der Proteine in Familien und Superfamilien. Beide Aspekte werden nun genauer betrachtet.

[5] Eine *community* ist eine Personengruppe, die eine Gemeinsamkeit aufweist, die diese Gruppe von der übrigen Gesellschaft abgrenzt ([BS98], S. 335f).

Annotation von Proteinsequenzen

Das Gruppenziel der Annotationsgruppe ist eine möglichst vollständige, redundanzfreie und qualitativ hochwertig annotierte Proteinsequenzdatenbank zu unterhalten. Um dieses Ziel erreichen zu können, müssen folgende Teilaufgaben von den Gruppenmitgliedern bewältigt werden:

- *Eingabe:* Auffinden und Einfügen neuer Proteinsequenzen in den Gruppeninformationsraum;
- *Pflege:* Pflege der bereits im Gruppeninformationsraum enthaltenen Proteinsequenzen mit ihren biologischen Zusatzinformationen (Annotationen);
- *Präsentation:* Bereitstellen dieser Informationen der wissenschaftlichen Öffentlichkeit.

Eingabe: In den ersten Jahren nach Gründung von *PIR-International* waren Publikationen von Proteinsequenzen in wissenschaftlichen Journalen primäre Quelle für neue Einträge in die Datenbank. Es wurden wissenschaftliche Artikel fotokopiert und die publizierte Primärsequenz von bis zu drei Personen unabhängig voneinander abgetippt, um Fehler möglichst auszuschließen. Anschließend erfolgte die manuelle Annotation der Proteinsequenz durch Exzerpieren des Artikels, in dem die Sequenz abgedruckt wurde.

Diese Vorgehensweise ist nicht mehr durchführbar. Verbesserte Sequenzierungstechniken erlauben heute systematische Sequenzierungen ganzer Genome in vergleichsweise kurzer Zeit. Die Sequenzierung des menschlichen Genoms mit seinen ca. drei Milliarden Basenpaaren ist seit einiger Zeit ein in den USA durch öffentliche Mittel gefördertes Projekt, das im Jahr 2003 abgeschlossen sein soll ([Wad98]). Es entstehen zunehmend Firmen, die ein kommerzielles Interesse an der Sequenzierung bestimmter Organismen und den damit verbundenen Informationen haben. So will z.B. die amerikanische Firma *Celera Genomics* ebenfalls das menschliche Genom entschlüsseln, allerdings innerhalb von zweieinhalb Jahren (bis zum Dezember 2001) und damit den öffentlich geförderten Projekten zuvorkommen ([Mar99]).

Aufgrund dieses gewaltigen Zuwachses an primärer Information, aus der Proteinsequenzen abgeleitet und annotiert werden müssen, ist eine rein manuelle Bearbeitung von Proteindaten nicht mehr durchführbar. Als Quellen neuer Proteinsequenzen dienen daher in erster Linie Nukleinsäuredatenbanken (siehe 2.4.1) sowie systematische Sequenzierungsprojekte (siehe 3.2.3), aus denen Daten direkt bei der Proteinsequenzdatenbank eingereicht werden (*direct submissions*).

Forderung: *Bereitstellen automatisierter Mechanismen, die Gruppenmitgliedern neue Proteinsequenzen, die bisher nicht Bestandteil des Gruppeninformations-*

raums sind, anzeigen und ggf. automatisiert in den Gruppeninformationsraum importieren.

Wird in einer Quelle eine neue Proteinsequenz gefunden, wird ein neues Proteinobjekt im Informationsraum der Proteinsequenzdatenbank erzeugt. Es wird ein Archiv von Proteinsequenzen unterhalten, in das jede neu in den Informationsraum importierte Proteinsequenz ohne Annotation abgelegt wird. Lediglich die Quelle, aus der sie extrahiert wurde, wird mit angegeben. Einträge im Archiv werden weder modifiziert noch gelöscht.

Forderung: *Realisierung eines Archivs für Proteinsequenzen.*

Das im Informationsraum neu erzeugte Proteinobjekt, eine zunächst rudimentäre Informationseinheit, wird in einem nächsten Schritt mit biologischer Zusatzinformation angereichert. Ziel ist, möglichst alle biologisch relevanten Informationen einzufügen, ohne die inhaltliche Konsistenz der Objekte des Informationsraums zu verletzen. In dieser ersten Phase der Annotation eines neuen Objekts sollen Informationsuntereinheiten, die in externen Quellen verfügbar sind, automatisiert übernommen werden. Dabei ist zu beachten, daß sich Quelle und Ziel hinsichtlich der Organisation biologischer Informationen unterscheiden können. Es sind geeignete Vorschriften aufzustellen, zu modellieren und umzusetzen, die einen automatisierten Import inhaltlicher Informationen gemäß der jeweiligen Organisation ermöglichen.

Forderung: *Unterstützung in der primären, automatisierten Annotation eines neuen Proteinobjekts durch Integration der in Quellen vorhandenen Informationseinheiten in das neue Proteinobjekt unter Beachtung aufgestellter Konvertierungsregeln.*

Ist die biologische Information der Quelle nicht umfassend oder nicht vertrauenswürdig genug, bzw. ist aufgrund der komplexen Semantik eine automatisierte Informationskonvertierung nicht durchführbar, ist diese beschriebene erste Phase der Annotation nicht ausreichend. In diesem Fall ist das Ergebnis ein annotiertes Objekt, das den qualitativen Ansprüchen der Gruppe nicht genügt. Es muß sich eine zweite, manuelle Phase anschließen, die durch die Bereitstellung von Werkzeugen der molekularbiologischen Sequenzdatenanalyse unterstützt wird (z.B. automatisiertes Erstellen von Genmodellen, Berechnung von Alignments ähnlicher Sequenzen in ausgewählten Organismen, etc.). Die Anwendung dieser Werkzeuge führt zu einer Verbesserung der Annotation. Es sind z.B. biologisch verwandte Proteinsequenzen zu finden, um evtl. dort enthaltene Informationen in das neue Objekt übernehmen zu können. Evtl. dienen diese auch als Ausgangspunkt für weitere Analysen. Zur Realisierung dieser zweiten Phase müssen die notwendigen Werkzeuge zur Verfügung gestellt werden. Es muß die Möglichkeit vorgesehen werden, neu entwickelte Werkzeuge leicht in das bestehende System integrieren

zu können. Das wissenschaftliche Personal muß über eine geeignete Infrastruktur verfügen, durch die zum einen Werkzeuge leicht angewandt und deren Ergebnisse automatisiert in die Annotationen einfließen können, zum anderen eine asynchrone, nebenläufige Bearbeitung von Objekten aus dem gemeinsamen Informationsraum ermöglicht wird.

Forderung: *Unterstützung in der zweiten, semi-automatisierten Annotation eines Proteinobjekts durch Bereitstellung von Werkzeugen zur molekularbiologischen Sequenzdatenanalyse sowie der Unterstützung im konkurrierenden Zugriff auf Objekte im Gruppeninformationsraum.*

Wartung: Die im Gruppeninformationsraum enthaltenen Annotationen der Proteinobjekte müssen ständig an den aktuellen Wissensstand angepaßt werden. Z.B. kann das Importieren neuer Objekte und damit potentiell das Integrieren neuer biologischer Sachverhalte die Annotationen der bereits in der Datenbank enthaltenen Objekte beeinflußen.

Forderung: *Unterstützung in der asynchronen nebenläufigen Bearbeitung von Proteinsequenzen zur Sicherstellung eines vollständigen, redundanzfreien und konsistenten Gruppeninformationsraums.*

Insbesondere kann die Pflege bedeuten, daß zwei Informationseinheiten zu einer einzelnen Informationseinheit verschmolzen werden (*merge*). Beispielsweise erfolgt zunächst die Publikation der Sequenz eines Gens. Zu einem späteren Zeitpunkt wird die Sequenz des gesamten Genoms publiziert, das die Sequenz des zuvor publizierten Gens enthält. Im Zuge der Aktualisierung der Proteinsequenzdatenbank hat diese Situation zur Folge, daß die gleiche Gensequenz redundant im Datenraum der Proteinsequenzdatenbank enthalten wäre. Daher werden in diesem Fall die beiden Einträge zu einem Eintrag zusammengefaßt.

Eine Überwachung und Sicherstellung der Aktualität kann nicht manuell durchgeführt werden. Es müssen Mechanismen realisiert werden, veraltete Informationen erkennen und anzeigen zu können, um Annotatoren eine geeignete Aktion zu ermöglichen. Neben der Wartung der biologischen Inhalte ist auch das eingesetzte kontrollierte Vokabular zu pflegen. Die Änderung der Schreibweise eines Schlüsselwortes z.B. muß eine automatisierte Korrektur aller Vorkommen dieses Schlüsselwortes in den Annotationen der Proteinobjekte nach sich ziehen. Neue Schlüsselwörter werden eingeführt. Proteine werden hinsichtlich ihrer Funktionen gemäß eines Kataloges klassifiziert. Dieser Katalog muß entsprechend dem zunehmenden Wissen bzw. aufgrund neuer Projekte, die bisher nicht behandelte Gebiete abdecken, aktualisiert werden.

Forderung: *Automatisiertes Überprüfen der Aktualität und Konsistenz innerhalb des Gruppeninformationsraums vorhandener biologischer Annotationen.*

Präsentation: Die in einer Proteinsequenzdatenbank enthaltenen biologischen Informationen müssen der wissenschaftlichen Allgemeinheit zur Verfügung gestellt werden. Es existieren zwei Arten der Informationspräsentation: bei der *statischen* Bereitstellung wird zu einem bestimmten Zeitpunkt der Datensatz eingefroren und für die Verbreitung entsprechend aufbereitet. Interessenten können dann via `ftp` Daten über das Internet auf lokale Rechner kopieren oder eine vom Datenbankbetreiber produzierte CD-ROM erhalten.

Daneben existieren *dynamische* Möglichkeiten, direkt auf den neuesten Stand der Informationseinheiten zugreifen zu können. Der interaktive Zugriff basiert auf den im *World-Wide Web* etablierten Protokollen (`http`) und Techniken (*Common Gateway Interface (CGI), applets*). Es werden geeignete graphische Darstellungen angeboten, die komplexe biologische Sachverhalte übersichtlich präsentieren.

Interaktivität ist eine Einschränkung vor dem Hintergrund umfassender Analysen oder Recherchen. Es muß die Möglichkeit angeboten werden, aus einem Prozeß heraus systematisch Informationsressourcen nutzen zu können, die über ein Netzwerk erreichbar sind. Der transparente Zugriff auf entfernte Objekte eines Informationsraums soll durch den Einsatz standardisierter Kommunikationsstrukturen ermöglicht werden.

Forderung: *Eine möglichst breite Palette von Verfahren zur Präsentation bzw. Techniken zum direkten oder indirekten Zugriff auf biologische Informationseinheiten im Gruppeninformationsraum.*

Durch diese Bandbreite an Zugriffsmöglichkeiten wird ein möglichst großer Interessentenkreis erreicht.

Klassifikation von Proteinen

Um die große Menge komplexer biologischer Informationen übersichtlich strukturieren zu können, werden Proteinsequenzen klassifiziert. Biologisch ähnliche Sequenzen[6] werden zu Familien zusammengefaßt. Dabei darf eine Sequenz nur genau einer Familie angehören, d.h. die Mengen der Familien sind disjunkt. Ferner werden Familien zu Superfamilien zusammengefaßt. Auch hier gilt, daß eine Familie in nur genau einer Superfamilie enthalten sein darf.

Wird zwischen zwei Sequenzen ein enger evolutionärer Zusammenhang angenommen, werden sie in die gleiche Familie klassifiziert. Dadurch wird der Raum der möglichen Funktionalitäten eines Proteins in der Zelle auf die Vertreter dieser Familie eingeschränkt. Wurden zu einem Familienmitglied experimentelle Resultate ermittelt, können diese Ergebnisse evtl. auf die Familienmitglieder übertragen

[6] Das bedeutet hier Ähnlichkeit auf Sequenzebene, die auch als Sequenzhomologie bezeichnet wird.

werden. Dadurch können globale Annotationen der Familie, d.h. die Summe aller Einzelannotationen aller Mitglieder, deutlich verbessert werden.

Allerdings ist bei jeder Form der automatisierten Analyse zu beachten, daß es in der Natur viele Ausnahmen gibt, die nicht durch allgemeine Algorithmen behandelt werden können. Die Expertise wissenschaftlicher Spezialisten ist nicht ersetzbar. Vielmehr müssen sie geeignet unterstützt werden, um ihr Wissen in den Informationsraum einbringen zu können. Die Strategie ist daher, Standardfälle automatisiert und lediglich Sonderfälle manuell zu behandeln.[7]

Diese Klassifikationen müssen durchgeführt und verwaltet werden. Aufgrund der rechnerisch aufwendigen Verfahren zur Ähnlichkeitsbestimmung zweier Proteinsequenzen (z.B. BLAST [AGM+90], FASTA [Pea90]) müssen inkrementelle Verfahren zum Einsatz kommen. Lediglich neue oder modifizierte Proteinsequenzen werden dabei behandelt. Das Einfügen einer neuen Proteinsequenz kann zur Folge haben, daß neue Proteinfamilien entstehen oder existierende Familien aufgeteilt werden müssen. Wurde in eine Familie ein neuer Vertreter eingefügt bzw. die Sequenz eines Familienmitglieds modifiziert, muß die Konsistenz der Familie und der entsprechenden Superfamilie überprüft werden.

Forderung: *Unterstützung in der Pflege von Proteinklassifikationen unter Einsatz inkrementeller Aktualisierungsverfahren.*

Da Klassifikationen eine strukturierte Sicht auf die Objekte des Informationsraums erlauben, sind diese Informationen auf einer, abstrakt gesehen höheren Ebene anzusiedeln, als die eigentlichen Proteinobjekte. Die Assoziation zwischen diesen beiden Ebenen muß in beide Richtungen modelliert sein: vom Familienobjekt zum Proteinobjekt sowie vom Proteinobjekt zum Familienobjekt.

Forderung: *Konzeptionelle Trennung von Proteindaten und assoziierten Informationen in einer abstrakten Hierarchie unter Beachtung der beiderseitigen Assoziationen.*

3.2.3 Systematische Genomsequenzierungsprojekte

Im Rahmen eines Genomsequenzierungsprojektes wird ein bestimmter Organismus von einem Laboratorium bzw. einem Konsortium von Laboratorien, die sowohl akademischen als auch kommerziellen Hintergrund besitzen können, systematisch sequenziert. Neben den beteiligten Laboratorien existiert in solchen Projekten als zusätzlicher Partner ein *Informatikkoordinator*. Die anfallenden Rohdaten müssen vom Informatikkoordinator geprüft, ggf. konvertiert, in Zusammenhang mit Daten anderer Laboratorien gebracht, annotiert und präsentiert werden.

[7] Sonderfälle sind nicht immer als solche erkennbar. Fehlerhafte, automatisiert durchgeführte Bearbeitungen müssen daher im Laufe der Zeit erkannt und von Experten korrigiert werden.

Desweiteren muß der Informatikkoordinator einen Überblick über den Status des Sequenzierungsschrittes haben. Die beteiligten Partner können sich innerhalb eines Landes befinden oder über ganz Europa verteilt sein. Selbst Projekte über Kontinentgrenzen hinweg sind keine Seltenheit. In den letzten Jahren wurden die kompletten Genome von *S. cerevisiae, C. elegans* und *A. thaliana* in internationalen Projekten ermittelt. Diese Verteilung der beteiligten Gruppen in ggf. unterschiedliche Zeitzonen muß bei der Unterstützung durch das Groupwaresystem beachtet werden.

Jedes Laboratorium sequenziert einen zuvor festgelegten Teilbereich des Genoms. Anschließend müssen diese Teile zum gesamten Genom assembliert werden. Im Rahmen der Sequenzierung wird DNS willkürlich in kleine, sequenzierbare Teilstücke zerlegt und die Basenfolge im Labor unter Einsatz von Sequenzierrobotern bestimmt. Die ermittelten Sequenzen werden auf elektronischem Weg an den Informatikkoordinator gesandt.

Forderung: *Übertragung sequenzierter DNS-Bereiche von Laboratorien an den Informatikkoordinator sowie Analyse und Verwaltung dieser Daten beim Koordinator.*

Aus den sequenzierten Fragmenten werden aufgrund existierender Fragmentüberlappungen im Rechner Fragmente zu längeren Teilstücken zusammengefaßt, bis schließlich die komplette Basenfolge des Genoms vorliegt.

Es wurden Algorithmen zur Erstellung von Genmodellen entwickelt, die Vorhersagen über das Vorhandensein von potentiellen Genen auf einem Chromosom treffen (siehe z.B. [FMMG98], [MT98]). Diese von verschiedenen Algorithmen erzeugten Genmodelle müssen von den Informatikkoordinatoren erstellt, verglichen und verwaltet werden. Ähnlich den Werkzeugen für die Annotation der Proteinsequenzdatenbank zur molekularen Sequenzdatenanalyse sind diese Algorithmen als Werkzeuge für die Gruppe der Informatikkoordinatoren in Genomsequenzierungsprojekten anzusehen. Eine entsprechende Integration in das System ist daher erforderlich.

Forderung: *Erstellen, Vergleichen und Verwalten von Genmodellen unter Anwendung spezialisierter Algorithmen, die als Werkzeuge in das System integriert werden müssen.*

Die genetischen Elemente, die auf der assemblierten DNS gefunden oder vorhergesagt werden, müssen annotiert werden. Dieser Vorgang ist vergleichbar mit der Annotation von Proteinsequenzen. Allerdings verfügt DNS neben den eigentlichen Genen über weitere Bereiche, wie z.B. regulatorische Elemente, die erkannt und annotiert werden müssen. Im Gegensatz zur Wartung einer Proteinsequenzdatenbank müssen in einem Genomprojekt Kontextinformationen verwaltet werden. Es ist nicht nur von Interesse, daß ein bestimmtes Gen und somit Protein als Pro-

dukt vorhanden ist, sondern auch seine genaue Position im Genom sowie evtl. gefundene regulatorische Einheiten in seiner Nähe. Bei höher organisierten Lebewesen müssen die nicht-kodierenden Abschnitte eines Gens verwaltet werden: es muß Intron/Exon Modelle geben. Um die Komplexität und Menge an Daten unter Wahrung des Überblicks manuell verwalten zu können, ist eine entsprechende Unterstützung durch graphische Darstellungen unumgänglich.

Forderung: *Unterstützung der Annotation von Genmodellen unter Zuhilfenahme graphischer Darstellungen.*

Die im Rahmen eines Genomprojektes gewonnenen Informationen, Sequenzdaten und ihre biologischen Annotationen, müssen den Projektpartnern sowie, evtl. zu einem späteren Zeitpunkt, der wissenschaftlichen Öffentlichkeit zugänglich gemacht werden. Diese Präsentationen können statisch in Form wissenschaftlicher Publikationen in Zeitschriften sowie statisch und dynamisch über das World-Wide Web erfolgen. Es müssen Mechanismen realisiert werden, die aufbauend auf dem Datenmanagement die textuelle und graphisch aufbereitete Präsentation mit der Möglichkeit der Interaktion durch den interessierten Anwender erlauben.

Forderung: *Automatisierte textuelle und graphische Präsentation der biologischen Sachverhalte mit der Möglichkeit der interaktiven Exploration.*

Wie bereits erwähnt, dienen die Nukleinsäuredatenbanken als Archiv für Primärsequenzen. Hat die Annotation einen gewissen Status erreicht, müssen die Sequenz sowie die assoziierte biologische Zusatzinformation im Rahmen eines Sequenzierungsprojektes als eine besondere Form der Veröffentlichung bei einem Nukleinsäuredatenbankbetreiber eingereicht werden, d.h. vom lokalen Gruppeninformationsraum in den globalen Informationsraum überführt werden. Dazu müssen die Daten gemäß den Forderungen des Datenbankbetreibers konvertiert werden. Es muß eine semantische Umsetzung der lokal vorhandenen Informationen im lokalen Modell in das externe Datenbankformat durchgeführt werden. Dieser Vorgang muß, nach Spezifikation der semantischen Umsetzung, automatisiert erfolgen.

Wurde eine Sequenz eingereicht, wird ein eindeutiger Schlüssel für diesen Eintrag seitens des Datenbankbetreibers als Ergebnis zurückgegeben. Es müssen Methoden realisiert werden, die Aktualisierungen (*updates*) an bereits eingereichten Sequenzen unter Verwendung dieses extern vergebenen Datenbankschlüssels ermöglichen. Dazu muß dieser Schlüssel in den lokalen Datensatz eingearbeitet werden.

Forderung: *Automatisiertes Einreichen bzw. Aktualisieren von Genomdaten bei öffentlichen Nukleinsäuredatenbanken als Form der Veröffentlichung.*

3.2.4 Analyse kompletter Genome

Das Interesse an der systematischen Analyse von Sequenzdaten entstand, als aufgrund der effizienten DNS-Sequenzierung der erste Organismus zu Beginn dieses Jahrzehnts systematisch komplett sequenziert wurde. Die zu dieser Zeit bekannten Proteine wurden in einem großen verteilten System analysiert: jeder individuelle Wissenschaftler hat Analysen durchgeführt und Proteinsequenzen samt Analyseergebnissen bei einer großen öffentlichen Datenbank eingereicht. In den ersten abgeschlossenen systematischen Sequenzierungsprojekten[8] wurden ca. 80000 Proteinsequenzen ermittelt. Die daraus entstandene Forderung, große Mengen an Sequenzdaten möglichst gleichzeitig bearbeiten zu können, ist mit einer manuellen Analyse nicht länger durchführbar. Dabei ist jedoch nicht nur die Zunahme des Datenvolumens als Grund zu sehen. Die Verfügbarkeit repräsentativer genomischer Fragmente (Chromosomen) bzw. kompletter Genome hat die Sequenzdatenanalyse substantiell verändert. Es sind eine Reihe neuer Aufgaben für die rechnergestützte Sequenzdatenanalyse entstanden:

- Entwicklung möglichst vollständiger Funktionskataloge von Genprodukten, die zum einen experimentell belegt sind, zum anderen vorhergesagt werden;
- Untersuchung der allgemeinen Organisation von Genen (wie z.B. die Genreihenfolge);
- Ableiten paraloger[9] Beziehungen in einem komplett sequenzierten Organismus durch eine systematische jeder-gegen-jeden Analyse der Genprodukte; dadurch ist eine Bewertung von Redundanz genetischer Information möglich;
- Ziehen von Schlußfolgerungen aufgrund des Fehlens bestimmter Proteine (und damit Funktionen) in der Zelle. Ist bekannt, daß ein Organismus eine bestimmte Funktion in der Zelle dennoch ausführt, ist dies ein Hinweis, daß diese Funktion von einem anderen Protein übernommen wurde. Nichtorthologe[10] Genersetzungen können auf diese Weise dokumentiert werden;
- Untersuchung der Zusammenhänge zwischen den Genprodukten pathogener Organismen und dem gesamten Spektrum bekannter Proteine; liefert wichtige Hinweise für die Entwicklung von Medikamenten und Impfstoffen;

[8] Bis März 2000 wurden 29 Genome systematisch sequenziert.

[9] *Paraloge* Gene sind homologe Gene innerhalb eines Organismus

[10] Bei *orthologen* Genen wird ein gemeinsamer Vorfahre angenommen. Ein Gen wurde im Laufe der Evolution an spezialisierte Organismen weitergegeben, ohne Veränderung der Funktion des Proteins. In diesen Orgnismen können sie als homologe Gene gefunden werden.

- Einsichten in organismusspezifische Eigenheiten aufgrund von inter-Genomvergleichen.

Forderung: *Automatisierte Analyse von Daten komplett sequenzierter Genome unter Einsatz spezialisierter Analysealgorithmen und unter Beachtung der aufgeführten Aspekte hinsichtlich der gewünschten Aussagen.*

Es wurden in den letzten Jahren Systeme entwickelt, die automatisiert große Mengen von Sequenzdaten vor diesem Hintergrund analysieren. Eine Darstellung und Bewertung dieser Systeme erfolgt in 3.5.

3.2.5 Systematische Funktionsanalyseprojekte

Funktionsanalyseprojekte beschäftigen sich mit der Ermittlung von Proteinfunktionen in lebenden Zellen. Um diese systematisch durchführen zu können, muß die gesamte DNS-Sequenz eines Organismus bekannt sein. Systematische Funktionsanalyseprojekte schließen sich daher i.d.R. an systematische Genomsequenzierungsprojekte an.

In einem Konsortium bestehend aus Laboratorien und Koordinatoren werden systematische Experimente durchgeführt. Die Ergebnisse einzelner Laboratorien werden von einem Informatikkoordinator gesammelt und verwaltet. Innerhalb des Konsortiums können Teilgruppen existieren, die sich auf bestimmte Aspekte der Funktionsanalyse (z.B. phänotypische Analyse, Expressionsanalyse auf RNA- und Proteinebene, metabolische Regulierung, Genstruktur, Beziehungen zu anderen Organismen) spezialisieren. Ein auf das Teilgebiet, in dem das Experiment durchgeführt wurde, spezialisierter Koordinator übernimmt die Kontrolle der biologischen Inhalte. Neben den fachspezifischen Koordinator und dem Informatikkoordinator existiert zusätzlich eine projektweite Sammelstelle, bei der biologische Einheiten (Stämme) der experimentellen Versuche des Projekts hinterlegt und dadurch dokumentiert werden. Der Zugriff auf einen bestimmten Stamm ist jederzeit möglich, falls weitergehende Versuche durchgeführt werden sollen.

Der Vorteil systematischer Funktionsanalyseprojekte liegt in der Belegbarkeit biologischer Aussagen durch wissenschaftliche Experimente. Diese Projekte können, da sie auf einem komplett sequenzierten Organismus durchgeführt werden, umfassende Aussagen über den Organismus erzeugen. Die erzielten Informationen sind aufgrund experimenteller Verifikationen als qualitativ hochwertiger anzusehen als Vorhersagen durch Algorithmen. Der Nachteil liegt darin, daß sie im Gegensatz zur Anwendung von Analysealgorithmen wesentlich aufwendiger durchzuführen und deutlich kostenintensiver sind.

Wie bei der Unterstützung in Genomsequenzierungsprojekten müssen Daten, die von Laboratorien erzeugt werden, von einem Informatikkoordinator gesammelt, verwaltet und, in einem geringeren Maß als bei Sequenzierungsprojekten,

annotiert werden. Es müssen Daten vom Laboratorium zum Informatikkoordinator unter Verwendung von Übertragungseinheiten über Kommunikationskanäle transferiert werden. Dabei müssen Formatbarrieren aufgrund des Einsatzes unterschiedlicher Programme bei den verschiedenen Partnern überwunden werden.

Forderung: *Möglichkeit der Übertragung experimenteller Daten von einem Laboratorium zum Informatikkoordinator.*

Beim Informatikkoordinator müssen die erzielten experimentellen Ergebnisse geeignet verwaltet werden. Dabei kann jedes Laboratorium seine Daten individuell, je nach Art des Experiments lokal erfassen. Das hat zur Folge, daß beim Informatikkoordinator eine Konvertierung der eingehenden Daten in das beim Koordinator bestehende Modell automatisiert durchgeführt werden muß. Aufgrund der komplizierten Semantik der biologischen Inhalte sind Regeln für Konvertierungen zu Beginn eines Projekts im direkten Gespräch zwischen den beteiligten Partnern festzulegen.

Forderung: *Datenkonvertierungen gemäß aufgestellter Regeln in eine einheitliche Repräsentation.*

Für den Zugriff auf die im Laufe des Projekts gewonnenen Informationen sind verschiedene Ebenen der Vertraulichkeit zu beachten:

- Auf unterster Ebene ist ein Zugriff lediglich einer Teilmenge aller beteiligten Partner gestattet, die z.B. einen besonderen Teilaspekt (wie etwa Expressionsanalyse) innerhalb des globalen Projekts verfolgen.
- Auf der nächsten Stufe ist der Zugriff allen Projektteilnehmern gestattet.
- Die Veröffentlichung der Daten stellt die oberste Ebene dar.

Zur Wahrung der jeweiligen Vertraulichkeit müssen geeignete Schutzmechanismen eingesetzt werden. Diese betreffen nicht nur die Datenhaltung, sondern auch die Datenübermittlung. Nicht nur das Abhören von Daten muß ausgeschlossen werden, sondern auch das kontrollierte oder unkontrollierte Manipulieren von Informationen.[11]

Forderung: *Realisierung von verschiedenen Ebenen der Vertraulichkeit unter Einsatz von Sicherheitsmechanismen zum Schutz gegen unauthorisierten lesenden bzw. schreibenden Zugriff.*

[11] Die Modifikation auch nur eines Zeichens in einer Proteinsequenz kann zu einer wertlosen Information führen, von der ggf. weitere, dann evtl. falsche Informationen abgeleitet und auf andere Proteinsequenzen übertragen werden.

3.3 Anwendungsszenarien

Im letzten Abschnitt wurden allgemeine Forderungen an ein Softwaresystem zur Unterstützung der jeweiligen Gruppen im Rahmen der Gen- und Genomanalyse dargestellt. Bei allen beschriebenen Projekten sind mindestens zwei Personen beteiligt, die gemeinsam versuchen, das jeweilige Gruppenziel zu erreichen. Aufgrund der Menge an Anforderungen und der Komplexität dieses Gebietes, werden im folgenden aus den vorgestellten Bereichen exemplarisch Teile herausgegriffen, für die ein Groupwaresystem entworfen, prototypisch umgesetzt, in der täglichen Arbeit eingesetzt und die Akzeptanz des Systems untersucht werden soll.

Vor diesem Hintergrund mußte anhand der aufgestellten allgemeinen Anforderungen an das CSCW-System bei der Modellierung beachtet werden, daß die in der prototypischen Umsetzung nicht unterstützten Gruppen zu einem späteren Zeitpunkt inkrementell in das geschaffene System integriert werden können. Auf Wiederverwendbarkeit bzw. Entwurf und Realisierung einer offenen Architektur wurde daher ein besonderes Augenmerk gelegt.

Der Schwerpunkt in der Unterstützung betrifft die Wartung der Proteinsequenzdatenbank, insbesondere unter den Aspekten Datenkonsistenz, Quantitäts- und Qualitätsverbesserung durch indirekte Kommunikationsunterstützung. Bei dieser Gruppe sind fast alle Teilaspekte der übrigen dargestellten Gruppen enthalten: Eingabe, Pflege und Präsentation. Für die Auswahl weiterer Bereiche zur Unterstützung durch das Groupwaresystem wurden die vorhandenen Gruppen auf gemeinsame Teilbereiche hin analysiert, die direkt eingesetzt werden können. Dadurch sollte ein möglichst breites Spektrum an Gruppen durch Softwareunterstützung partizipieren können, ohne zu spezielle, auf bestimmte Untergruppen zugeschnittene Anforderungen umsetzen zu müssen.

Zu allen wissenschaftlichen Aussagen, die über einen Sachverhalt getroffen werden, ist die Angabe der Quelle, woher die jeweilige Information stammt, essentiell. Diese Anforderung besteht bei allen aufgeführten Gruppen in diesem Anwendungsbereich. Daher wurde als weiterer Aspekt für die Unterstützung durch das Groupwaresystem die Verwaltung wissenschaftlicher Literatur ausgewählt.

Neben großen öffentlichen Nukleinsäuredatenbanken sind systematische Sequenzierungsprojekte eine wichtige Quelle für neue Proteinobjekte im globalen Datenraum der Proteinsequenzdatenbank. Da in der wissenschaftlichen Arbeitsgruppe, in der das System zum Einsatz kommen sollte, systematische Genomsequenzierungsprojekte betreut werden, ist als weitere Umsetzung die vollkommen automatisierte Einarbeitung neuer Proteinsequenzen aus diesen Sequenzierungsprojekten in die Proteinsequenzsammlung zu realisieren. Dadurch wird die implizite Kommunikation zwischen unterschiedlichen Untergruppen einer Arbeitsgruppe realisiert.

Um die Anforderungen zur Unterstützung oben genannter Teilbereiche erken-

nen und formalisieren zu können, werden Alltagsszenarien von Gruppenmitgliedern der jeweiligen Gruppen dargestellt. Davon wird ein konkreter Anforderungskatalog an ein CSCW-System abgeleitet.

Bevor konkrete Alltagsszenarien aufgezeigt werden, wird im folgenden die wissenschaftliche Arbeitsgruppe, in der das System zum Einsatz kam und kommt, kurz vorgestellt.

3.3.1 Die wissenschaftliche Arbeitsgruppe MIPS

Das *Münchener Informationszentrum für Proteinsequenzen (MIPS)*, eine Arbeitsgruppe der *Gesellschaft für Umwelt und Gesundheit (GSF)* am Max-Planck-Institut für Biochemie, Martinsried bei München, wurde 1988 durch ein Projekt des Bundesministeriums für Forschung und Technologie (BMFT, heute BMBF) mit der Maßgabe ins Leben gerufen, die internationalen Bestrebungen im Bereich der Proteinsequenzdatenbanken zu unterstützen ([MFG+00] [MHK+99] [MPH97] [KHMM96]). Seit seiner Gründung wird MIPS von H.-W. Mewes geleitet.[12] MIPS ist europäischer Partner von *PIR-International*, das in enger Kooperation mit Datenbankbetreibern in den USA und Japan eine Proteinsequenzdatenbank erstellt, europaweit auf CD-ROM vertreibt sowie über das WWW zugänglich macht.[13] In den letzten 10 Jahren wurden bei MIPS ca. 80.000 Proteinsequenzen prozessiert und annotiert.

Darüber hinaus ist MIPS im Bereich der molekularen Sequenzdatenanalyse tätig. In den Jahren 1989 bis 1996 fungierte MIPS als Informatikkoordinator des EU-Projekts zur Sequenzierung der Bäckerhefe *Saccharomyces cerevisiae*, das im April 1996 mit der Veröffentlichung des komplett sequenzierten Genoms erfolgreich beendet wurde ([GBB+96] [MAB+97] [KHF+97]). Derzeit werden die Projekte zur Sequenzierung der Pflanze *Arabidopsis thaliana* (EU-Projekt [MSW+99]) und des Pilzes *Neurospora crassa* (DFG-Projekt) sowie das europäische Projekt zur Funktionsanalyse der Bäckerhefe unterstützt. Weiterhin ist MIPS für die Bioinformatik in einem Projekt des BMBF zuständig, in dem vollständige menschliche cDNA-Sequenzen von einer Reihe von deutschen Sequenzierungslaboratorien aus dem akademischen und industriellen Umfeld sequenziert werden. Im deutschen Humangenomprojekt entwickelt MIPS eine aktive Datenbank. Ferner wurde ein System zur systematischen Analyse ganzer Genome entwickelt (*PEDANT*, [FM97a]).

Die bei MIPS durchgeführten Projekte sind aufgrund des großen Informatikanteils nur durch einen interdisziplinären Ansatz zu bewältigen: Biologen, Chemiker, Biochemiker und Informatiker arbeiten eng zusammen. Diese Zusammen-

[12] Email: mewes@mips.biochem.mpg.de

[13] http://www.mips.biochem.mpg.de

arbeit wird durch die örtliche Nähe (Kooperation innerhalb einer Arbeitsgruppe) vereinfacht. In dieser Gruppe wurde der Prototyp des hier beschriebenen CSCW-Systems produktiv eingesetzt.

3.3.2 Wartung einer Proteinsequenzdatenbank

In diesem Unterabschnitt werden konkrete Anforderungen aus der Gruppe dargestellt, die sich bei MIPS mit der Wartung der Proteinsequenzdatenbank *PIR-International* beschäftigt. MIPS ist innerhalb der Organisation von *PIR-International* für Europa zuständig. Weitere Partner befinden sich in den USA sowie Japan (siehe 2.4.2).

Zusammensetzung und Organisation der Gruppe

Diese Gruppe besteht aus einem Gruppenleiter sowie fünf Gruppenmitgliedern, mit einem hohen Anteil an Teilzeitarbeit. Ein großer Teil der Personen arbeitet zu Hause (*offline*) bzw. von zu Hause aus (*online*). Diese Aspekte der Telearbeit mit den damit verbundenen frei einteilbaren Arbeits- und Anwesenheitszeiten müssen beim Entwurf des Groupwaresystem geeignet unterstützt werden.

Jedes Gruppenmitglied verfügt an der Arbeitsstelle über einen Arbeitsplatzrechner (*workstation*) mit entweder Unix (Digital Unix, Linux) oder OpenVMS als Betriebssystem. Gruppenmitglieder müssen in die Lage versetzt werden können, Kopien der Objekte, die in den Verantwortungsbereich des jeweiligen Gruppenmitglieds fallen, auf privaten Rechnern zu Hause oder unterwegs bearbeiten zu können. Dabei sollen sowohl möglichst geringe Anforderungen an die eingesetzten Systeme gestellt, als auch möglichst viele verschiedene Systeme unterstützt werden.

Die Aufgabenzuteilung an einzelne Gruppenmitglieder erfolgt durch den Gruppenleiter, der Bearbeitungsaufträge an Annotatoren erteilt. Der Gruppenleiter benötigt dazu einen entsprechenden Überblick über potentielle Quellen neuer Proteinsequenzen sowie die Auslastung der Gruppenmitglieder, um geeignete Auftragsvergabestrategien anwenden zu können. Er muß über eine Infrastruktur verfügen, die die Delegation neuer Bearbeitungsaufträge an seine Gruppenmitglieder erlaubt.

Im CSCW-System wird jedem Gruppenmitglied ein individueller Bearbeitungsraum, der private Informationsraum, zugewiesen. Neu zu bearbeitende Einheiten werden vom Gruppenleiter dem Gruppenmitglied mitgeteilt. Das Gruppenmitglied besitzt die Freiheit, in seinem privaten Informationsraum enthaltene Aufträge zur Bearbeitung auszuwählen. Die Auftragsbearbeitung führt zur Generierung neuer Proteinobjekte, die zunächst im privaten Informationsraum des Annotators angelegt werden. Ist die Bearbeitung des Objekts abgeschlossen, wird

es in den gemeinsamen Informationsraum der Gruppe übernommen. Im privaten Informationsraum werden sowohl das Proteinobjekt als auch sein zugehöriger Bearbeitungsauftrag entfernt.

Der Gruppenleiter kann die Zustände der Bearbeitungsaufträge nur so weit überwachen, daß die Anzahl der noch zur Bearbeitung anstehenden Objekte bekanntgegeben werden, d.h. die Anzahl der noch nicht bearbeiteten Aufträge. Es muß ausgeschlossen sein, daß eine genauere Überwachung der Gruppenmitglieder und ihrer persönlichen Informationsräume durchgeführt werden kann, um eine Nicht-Akzeptanz des Systems durch die Gruppenmitglieder aufgrund mangelnder Privatsphäre zu vermeiden.

In private Informationsräume können Replikate von Objekten des gemeinsamen Gruppeninformationsraums eingebracht werden. Modifikationen der Inhalte biologischer Objekte sind nur auf diesen Replikaten durchführbar. Dem Gruppenmitglied ist es freigestellt, wie ein repliziertes Objekt bearbeitet wird. Es ist das Kopieren von Replikaten auf einen Datenträger und der anschließenden Bearbeitung außerhalb des privaten Informationraums durchführbar.[14] Auf diese Weise kann die Forderung nach Unterstützung von Telearbeit angeboten werden: entweder das Gruppenmitglied ist an seinem Arbeitsplatz und nimmt Replikate auf einem Datenträger oder Laptop mit nach Hause oder es führt die entsprechenden Arbeitsschritte von zu Hause aus durch. Replikate werden über das bestehende Netzwerk (ISDN-Verbindung) auf den heimischen Rechner übertragen.

Nach Abschluß der Bearbeitung wird das Replikat des Bearbeitungsraums wieder, unter Beachtung der Konsistenzforderung, in den gemeinsamen Informationsraum eingefügt. Die durchgeführten Änderungen am Ausgangsobjekt sind sofort für alle Gruppenmitglieder sichtbar. Wünschenswert ist in diesem Zusammenhang eine Verwaltung von Versionen des Objekts. Dadurch können durchgeführte Modifikationen nachvollzogen werden. Sämtliche Replikate des privaten Informationsraums werden entfernt. Eventuell noch existierende Replikate, die vom Gruppenmitglied eigenständig angefertigt wurden und sich außerhalb des Gruppeninformationsraums befinden, sind dem System nicht bekannt und können daher auch nicht automatisch gelöscht werden. Diese Kopien unterliegen der Verantwortung des Gruppenmitglieds. Das System muß jedoch sicherstellen, daß veraltete, nicht gelöschte Replikate nicht zu einem späteren Zeitpunkt wieder in den Gruppeninformationsraum übernommen werden können.

[14] Es wird an dieser Stelle ein Replikat eines Replikats erzeugt. Die dadurch entstehende Möglichkeit der Inkonsistenz zwischen den Replikaten wird durch das System nicht verhindert. Es obliegt vielmehr der Verantwortung eines jeden Gruppenmitglieds, die eigenständig angefertigten Replikate konsistent zu halten bzw. persönliche Versionsmanagementstrategien durchzuführen.

Integration neuer Proteinsequenzen

Gruppenmitglieder müssen im Rahmen der Pflege neue Proteine in den Gruppeninformationsraum einfügen. Wie bereits unter 3.2.2 dargestellt, ist das manuelle Eingeben neuer Proteinsequenzdaten mit anschließender manueller Annotation nicht mehr effektiv durchführbar. Hauptquellen neuer Proteindaten sind Nukleinsäuredatenbanken sowie systematische Sequenzierungsprojekte. Letztere erlauben das automatische Extrahieren neuer Proteinobjekte.

Für jede Form des Importierens neuer Informationen gilt folgende grundlegende Situation: Es liegt ein Wort einer quellenspezifischen Sprache (z.B. ein Nukleinsäuredatenbankeintrag der *EMBL Nucleotide Sequence Database*) vor, das in ein Wort der Senke (Proteinsequenzdatenbank) übersetzt werden muß. Für existierende Sprachen, die durch das System unterstützt werden sollen, müssen sprachenspezifische Übersetzer (*Compiler*) entwickelt werden. Diese translatieren Worte der Ausgangssprache in entsprechende Worte der Zielsprache (Proteinobjekte).

Liegt als Quelle ein Nukleinsäuredatenbankeintrag vor, erfolgt als erster Schritt eine automatisierte Annotationsphase, in der ein noch unvollständig annotiertes neues Proteinobjekt erzeugt wird. In dieser Phase werden unter Einsatz eines spezialisierten Übersetzers relevante Teilinhalte des Ausgangsobjekts in der Nukleinsäuredatenbank extrahiert und in ein neues Proteinobjekt integriert. Spezifizierte Regeln erlauben es, diese Informationen aus Untereinheiten eines Nukleinsäuredatenbankeintrages in das neu erzeugte Protein zu übernehmen.

Das Gruppenmitglied muß zu Beginn der Bearbeitung einer neuen Proteinsequenz festlegen, welcher Eintrag der Quelle bearbeitet werden soll. Jedes Gruppenmitglied verfügt dazu über einen individuellen Bearbeitungsraum, der Bearbeitungsaufträge enthält, die vom Gruppenleiter zugeteilt wurden. Bearbeitete Aufträge werden aus dem Bearbeitungsraum entfernt, neue Aufträge werden von dem Gruppenleiter eingefügt. Die Bearbeitungsräume sind partitioniert, d.h. Mehrfachbearbeitungen gleicher Quelleinheiten werden vermieden. Bei der Zuteilung von Bearbeitungsaufträgen an die jeweiligen Gruppenmitglieder kann der Gruppenleiter vorhandenes Spezialwissen bei Gruppenmitgliedern (z.B. über einen bestimmten Organismus oder bestimmte Pflanzenarten etc.) berücksichtigen. Das Gruppenmitglied kann jederzeit den Zustand des eigenen Bearbeitungsraums überprüfen. Wurde der letzte Bearbeitungsauftrag entfernt, wird der Gruppenleiter über dieses Ereignis informiert (Notifikation).

Durch das Auswählen eines Bearbeitungsauftrags im Bearbeitungsraum wird die erste Phase der Annotation gestartet. Es wird die entsprechende Einheit aus der Quelle extrahiert und der quellensprachenspezifische Übersetzer gestartet. Als Ergebnis liegt ein neues, rudimentär annotiertes Proteinobjekt im privaten Informationsraum des Gruppenmitglieds vor.

Nach dieser ersten, automatisierten Phase kann eine manuelle Weiterbearbei-

tung durch das Gruppenmitglied erfolgen. Eine manuelle Bearbeitung ist notwendig, wenn gewisse semantische Situationen im Ausgangswort nicht adäquat vom Übersetzer bzw. den zugrundeliegenden semantischen Regeln bearbeitet werden können (z.B. mehrere biologische Sachverhalte in einem Freitextbereich), oder biologische Informationseinheiten an die im gemeinsamen Datenraum existierenden Rahmenbedingungen angepaßt werden müssen (z.B. einheitliche Taxonomieinformationen). Nach Beendigung dieser manuellen Bearbeitung, durch die die automatisiert generierten Annotationen überschrieben werden können, liegt ein neues, vollständig annotiertes Proteinobjekt im persönlichen Informationsraum vor, das nach Abschluß der Import-Phase in den Gruppeninformationsraum migriert. Dieses Objekt steht sofort allen Gruppenmitgliedern zur Verfügung. Der Bearbeitungsraum sowie der private Informationsraum des Gruppenmitglieds wird entsprechend aktualisiert: der entsprechende Bearbeitungsauftrag sowie Kopien des Proteinobjekts werden entfernt.

Diese Partitionierung der Bearbeitungsaufträge, durch die genau eine Person für jeden Auftrag verantworlich ist und dadurch Konflikte vermieden werden, kann durch Interaktion zwischen Gruppenmitgliedern durchbrochen werden. Während der Bearbeitung muß es einem Gruppenmitglied ermöglicht werden, die Verantwortung für ein Proteinobjekt auf ein anderes Gruppenmitglied übertragen zu können (z.B. aufgrund speziellen Fachwissens). Dazu muß ein Proteinobjekt zwischen privaten Informationsräumen unter Wahrung der Konsistenz migrieren können. Das neu für das Objekt verantwortliche Gruppenmitglied kann nach Beendigung der Bearbeitungen das Objekt in den gemeinsamen Gruppeninformationsraum einbringen.

Dienen als Quelle für neue Proteinobjekte systematische Sequenzierungsprojekte, können neue Proteinobjekte vollkommen automatisiert erzeugt und in den gemeinsamen Informationsraum integriert werden. Wird die Sequenzierung von einer Untergruppe der gleichen Arbeitsgruppe betreut, sind entsprechende Absprachen, die semantischen Einheiten der jeweiligen Informationen betreffend, zwischen den beteiligten Untergruppen möglich. In diesem Fall kann eine manuelle Phase entfallen. Es muß jedoch, wie in der soeben beschriebenen Situation, ein quellensprachenspezifischer Übersetzer entwickelt werden.

Neben der Unterstützung der Annotationsgruppe durch eine vollautomatisierte Annotation, muß eine entsprechende Unterstützung für Gruppenmitglieder der Gruppe angeboten werden, die das Sequenzierungsprojekt betreuen. Nach einem ersten Abschluß der Annotationen eines bestimmten DNS-Abschnitts von Chromosom 4 der Pflanze *Arabidopsis thaliana*, wird diese Information von einem Gruppenmitglied der Gruppe, die das Sequenzierungsprojekt betreut, bei der Nukleinsäuredatenbank *EMBL Nucleotide Sequence Database* eingereicht. Gleichzeitig sollen neue vollautomatisiert annotierte Proteinobjekte für die Proteinsequenzdatenbank *PIR-International* erzeugt werden.

Es findet ein Informationstransfer zwischen Gruppengrenzen statt. Mitglieder der einen Gruppe erzeugen Objekte, die von Mitgliedern einer anderen Gruppe bearbeitet werden. Die Gruppenmitglieder der Proteinsequenzdatenbankgruppe erhalten entsprechende Notifikationen durch das System (indirekte Kommunikation zwischen Gruppenmitgliedern). Bei dieser vollautomatisierten Annotation erfolgen keine Eintragungen in Bearbeitungsräume oder private Informationsräume von Annotatoren der Proteinsequenzdatenbankgruppe.

Dieser direkte Import führt zu einer qualitativen Verbesserung, da neue Proteinobjekte früher in den Datensatz der Proteinsequenzdatenbank gelangen als über den Umweg externer Nukleinsäuredatenbanken.

Wartung existierender Proteinobjekte

Die im gemeinsamen Informationsraum enthaltenen biologischen Informationen müssen ständig an den aktuellen Wissensstand angepaßt werden. Darüberhinaus müssen im Rahmen von Wartungsarbeiten (z.B. Eintrag neuer Annotationsinformation) Änderungen an Proteinobjekten durchgeführt werden.

Objekte müssen zusammengefaßt werden können. Wurde z.B. zunächst lediglich ein Fragment einer Proteinsequenz in den Gruppeninformationsraum übernommen, sollen nach Import der kompletten Sequenz diese beiden Objekte zu einem Repräsentanten zusammengefaßt werden (*merge*). Dieser Mechanismus ist notwendig, um Redundanzfreiheit sicherstellen zu können.

Sperrmechanismen müssen gewährleisten, daß konkurrierende Zugriffe von Gruppenmitgliedern auf Objekte des gemeinsamen Informationsraums nicht zu Inkonsistenzen führen. Je nach Anwendung ist zu entscheiden, ob optimistische oder pessimistische Verfahren zur Nebenläufigkeitskontrolle zum Einsatz kommen sollen.

Um eine Modifikation auf einem Objekt durchführen zu können, muß das Gruppenmitglied das entsprechende Objekt aus dem gemeinsamen Informationsraum anfordern. Ist das entsprechende Objekt verfügbar, wird im privaten Informationsraum des Gruppenmitglieds ein Replikat erzeugt. Das System muß sicherstellen, daß aufgrund der notwendigen Kommunikation zwischen den zur Bearbeitung eines Änderungsauftrags beteiligten Objekten keine Inkonsistenzen aufgrund des Zeitverlustes auftreten können (für Details siehe Kapitel 4.8.4).

Kann ein Gruppenmitglied auf ein angefordertes Objekt nicht zugreifen, muß es über diesen Zustand in geeigneter Form informiert werden. Die minimale Rückkoppelung durch das System besteht im Informieren, daß ein schreibender Zugriff derzeit nicht möglich ist. Diese Information allein ist jedoch nicht ausreichend. Gruppenmitglieder müssen informiert werden, welches Gruppenmitglied seit welchem Zeitpunkt ein Objekt bearbeitet; die Telefonnummer dieser Person muß angegeben und das einfache Erstellen und Verschicken einer Email muß ermöglicht

werden. Soweit möglich wird erwähnt, ob und ggf. auf welchem Rechner (inklusive Standort des Rechners im Gebäude) das Gruppenmitglied derzeit im System angemeldet ist. Letztere Forderung ist besonders von Bedeutung, da die Arbeitsgruppe über zwei verschiedene Gebäude innerhalb eines Ortes verteilt ist. Durch diese breite Palette an Möglichkeiten, wie Gruppenmitglieder miteinander direkt in Kontakt treten können, wird die gruppeninterne Kommunikationsarbeit erleichtert. Es wird dem Gruppenmitglied außerdem der Eindruck vermittelt, daß es Bestandteil einer Gruppe ist und nicht allein mit dem System interagiert (*awareness*).

Präsentation und Bereitstellen der Informationen

Eine Informationsressource wird nur dann genutzt, wenn die enthaltenen Informationen entsprechend den Anforderungen der *community* angeboten und präsentiert werden. Es ist zu unterscheiden zwischen der Bereitstellung, d.h. wie kann ein externer Interessent Kopien der Informationen zur individuellen Weiterverarbeitung erhalten, und der Präsentation, d.h. wie werden die Inhalte dargestellt bzw. kann im öffentlichen Informationsraum recherchiert werden.

Im Bereich der Biotechnologie haben sich in der wissenschaftlichen Öffentlichkeit folgende Protokolle bzw. Technologien etabliert: um komplette Kopien von Datenbanken zu erhalten wird der Dienst `ftp` verwendet. Zusätzlich vertreiben Datenbankbetreiber Kopien ihrer Daten auf CD-ROM. Um einen entfernten Zugriff aus einem Prozeß heraus zu ermöglichen, wird zunehmend der Industriestandard *CORBA* eingesetzt.

Regelmäßig sollen Kopien der Proteinobjekte via `ftp` und CD-ROM angeboten werden. Dazu müssen alle Objekte in ein geeignetes Format konvertiert und in eine bzw. mehrere Textdateien exportiert werden.

Präsentationen der biologischen Inhalte erfolgen im *WWW*. Neben statischen HTML-Seiten wird zur Realisierung von Dynamik (z.B. Recherche im Informationsraum) die *CGI*-Technologie[15] eingesetzt. Zunehmend gewinnen *Java Applets* an Bedeutung, besonders in Kombination mit CORBA. Browsersoftware, wie etwa *Netscape*, enthalten bereits CORBA Elemente. Dadurch können sehr leicht Clients entwickelt werden, die innerhalb einer HTML-Seite ablaufend unter Verwendung der browserinternen CORBA Elemente auf entfernte Objekte zugreifen können. War es den weltweiten Datenbankbetreibern in diesem Gebiet der Biologie bisher nicht gelungen, ein einheitliches Datenaustauschformat auf Textebene zu definieren (alle Ansätze in diese Richtung scheiterten), ist durch den Einsatz von CORBA Interoperabilität zwischen unabhängigen Informationsanbietern möglich geworden.

Da der entfernte Objektzugriff nicht immer die ideale Lösung darstellt, z.B.

[15] *Common Gateway Interface*

wenn größere Datenmengen an einen Client übertragen werden müssen, findet vermehrt der Einsatz von *XML (eXtensible Markup Language*, siehe z.B. [Lau98]) an Bedeutung. Dabei werden Informationsrepräsentationen aus Objekten in spezialisierte Sprachen, die in XML abgefaßt sind, übersetzt und ausgetauscht.

3.3.3 Literaturverwaltung

Literatur muß bei allen wissenschaftlich durchgeführten Projekten verwaltet werden. Um im vorliegenden Fall zum einen Redundanz zu vermeiden, zum anderen einen auf die bei MIPS durchgeführten Projekte fokussierten Literaturdatenbestand zu erhalten, ist eine zentrale Literaturverwaltungskomponente zwingend notwendig. Diese Fokussierung des Datenbestandes führt bei durchgeführten Literaturrecherchen zu schnelleren Anfrageergebnissen sowie zu kleineren Treffermengen, die vom Anwender analysiert werden müssen. Daher wird eine lokale Literaturverwaltung einem externen Literaturdienst vorgezogen.

Um eine umfassende Recherche durchführen zu können, ist der Zugriff auf große Literaturdatenbanken, wie etwa MEDLINE, unabdingbar. Ergebnisse einer Recherche müssen leicht in den lokalen Literaturdatenbestand übernommen werden können. Lokale Objekte können auf diese Zitate verweisen. Das interaktive Aufsuchen eines solchen Verweises muß schnell zum Resultat führen. Diese Anforderung ist durch große Literaturdatenbankanbieter, die über das Internet erreichbar sind, nicht immer gewährleistet. Ein lokaler fokussierter Literaturdatenbestand bietet den Vorteil der besseren Verfügbarkeit.

Alle laufenden und anstehenden Projekte sollen diesen Literaturdienst nutzen. Er wird daher als zentraler Basisdienst für Gruppenmitglieder angesehen. Es müssen verschiedene Typen von Literaturstellen verwaltet werden: Buchzitate, Artikelreferenzen aus wissenschaftlichen Zeitschriften inklusive Zusammenfassung (*abstract*), direkte Einreichungen an Datensammlungen (*submissions*), etc. Alle biologischen Objekte verweisen, statt die entsprechende Literatur aggregiert zu enthalten, auf das jeweilige Literaturobjekt unter Verwendung eindeutiger Objekt-IDs (Schlüssel).

Eine Anforderung an die Literaturverwaltung ist die Sicherstellung von Redundanzfreiheit. Dazu müssen geeignete Regeln formuliert werden, um diese Forderung automatisiert sicherstellen zu können. Ein Artikel aus einer wissenschaftlichen Zeitschrift ist durch den Zeitschriftennamen, die Ausgabe der Zeitschrift, Seitennummer der ersten Seite des Artikels sowie die Autoren festgelegt.[16] Bei Fachzeitschriften können Zeitschriftennamen sehr lang sein. Die Verwendung von

[16]Die Möglichkeit, daß identische Autoren in der gleichen Ausgabe einer Zeitschrift auf der gleichen Seite zwei unterschiedliche Artikel veröffentlichen, hat in der Praxis keine Bedeutung und wird daher hier auch nicht näher betrachtet. Streng genommen kann diese Aufweichung zu Redundanzen im Informationsraum führen.

Abkürzungen ist daher bewährte Praxis. Allerdings sind diese Abkürzungen nicht standardisiert und es werden zu einem Journal eine Reihe von Variationen von verschiedenen Literaturdiensten verwendet. Die Abkürzung *Proc. Natl. Acad. Sci.* steht z.B. für *Proceedings of the National Academy of Sciences of the United States of America*. Dieses Journal wird gelegentlich auch mit *Proc Natl Acad Sci USA* abgekürzt. Damit in einem Prozeß entschieden werden kann, ob ein bestimmter Artikel bereits im Informationsraum enthalten ist, muß zunächst eine Konvertierung des Zeitschriftennamens in eine im Informationsraum einheitlich verwendete Schreibweise vorgenommen werden. Danach kann auf Redundanz geprüft werden. Es muß somit eine Verwaltungseinheit für Journalnamen inklusive ihrer jeweiligen Abkürzungen bereitgestellt und gewartet werden.

Es folgen eine Reihe von Szenarien, wie sie im Alltag der Gruppenmitglieder auftreten.

Exzerpt eines wissenschaftlichen Artikels

Bei der Durchsicht einer bestimmten, einen speziellen Organismus betreffenden wissenschaftlichen Zeitschrift, stößt eine Biologin, die das Sequenzierungsprojekt dieses Organismus betreut, auf einen Artikel, der wichtige Erkenntnisse über einen besonderen Aspekt des Organismus enthält. Diese Informationen werden von der Biologin sofort verarbeitet, indem entsprechende Modifikationen an den Annotationen der betroffenen Objekte durchgeführt werden. Die Quelle, in diesem Fall der vorliegende Artikel, wird mitsamt der Zusammenfassung des Artikels (*abstract*) in die Literaturverwaltung aufgenommen. Als Ergebnis wird von der Literaturverwaltung eine neue eindeutige Objekt-ID zurückgegeben.

Die neu integrierte biologische Information verweist auf das neue Literaturobjekt in der Literaturverwaltung unter Angabe der korrespondierenden Objekt-ID, d.h. ist in der Annotation lediglich als Schlüssel enthalten. Diese Objekt-IDs sollen vor Anwendern versteckt werden. In einer HTML-Seite z.B. können sich entsprechende Objekt-IDs hinter einem Querverweis (*link*) verbergen. Durch Verfolgen dieses Querverweises werden die im Querverweis als Parameter spezifizierten Objekt-IDs an die Literaturverwaltung weitergereicht, die die entsprechenden Literaturobjekte aus der Datenbank extrahiert. Diese Objekte können dem Anwender präsentiert werden. Die Modellierung der Literaturobjekte muß dabei so erfolgen, daß sich jedes Literaturobjekt in verschiedenen Formaten (wie z.B. HTML) darstellen kann. Als weitere Formate werden die zum Export benötigten Darstellungen unterstützt.

Systematische automatisierte Literaturrecherche

Für die Unterstützung im Rahmen eines Sequenzierungsprojektes wurde für einen externen Literaturdienst ein Profil erstellt, nach dem die entsprechenden wissenschaftlichen Quellen nach neuen Publikationen durchsucht werden. Regelmäßig eingehende Ergebnisse dieser Literaturrecherchen in Form von Emails sollen automatisiert in die Literaturverwaltung übernommen werden, falls das entsprechende Zitat noch nicht enthalten ist. Zusammenfassungen sollen dabei ebenfalls in das lokale Literaturobjekt eingetragen werden.

Dazu müssen Übersetzer entwickelt werden, die die Sprache, in denen Ergebnisse von Literaturdiensten verfaßt werden, verarbeiten können. Es erfolgt eine Konvertierung relevanter Informationseinheiten in neu generierte Literaturobjekte. Diese müssen unter Beachtung der Redundanzfreiheit in den gruppenweiten Literaturbestand aufgenommen werden.

Alle Schritte, vom Empfang der Ergebnisse bis zur Aufnahme neuer Zitate in die Literaturverwaltung, müssen automatisiert erfolgen.

Persönliche Literaturrecherchen

Literaturrecherchen, wie soeben dargestellt, werden von externen Literaturdiensten durchgeführt, die für ihre Dienste bezahlt werden müssen. Um systematische Literaturrecherchen in öffentlich zugänglichen Literaturdatenbanken zu unterstützen, sollen persönliche Recherchen durchführbar sein.

Ein Gruppenmitglied definiert ein Profil, z.B. durch Angabe einer Liste von Schlagwörtern. Regelmäßig ermittelt ein Literaturagent,[17] ob in den öffentlich zugänglichen Literaturdatenbanken neue Zitate existieren, die auf das Profil passen. In diesem Fall werden neue Zitate in eine, dem individuellen Gruppenmitglied assoziierte persönliche Literaturdatenbank eingefügt. Das Gruppenmitglied wird bei der Verwaltung dieser Literatur entsprechend unterstützt. Beim Anmelden werden neue Literaturstellen angezeigt. Das Gruppenmitglied kann manuell entscheiden, ob ein Zitat in den globalen Literaturbestand übernommen, zurückgestellt oder als uninteressant markiert werden soll. Letzterer Fall entspricht einem Löschen. Damit allerdings ein gelöschtes Zitat bei der nächsten Agentensuche nicht als neue Literaturstelle angeboten wird, muß dem Agenten die Information, daß dieses Zitat uninteressant war, zur Verfügung stehen.

Der Agent soll anhand der ausgewählten sowie der als uninteressant markierten Zitate das vom Gruppenmitglied aufgestellte Profil überprüfen und ggf. Verbesserungsvorschläge anbieten. Dadurch soll die Anzahl der falschen Zitate in

[17] Definition Agent: „Ein Agent ist ein autonom arbeitendes Programm, das gewisse Routinearbeiten für eine Person übernimmt und sozusagen als deren Stellvertreter fungiert.“ [Bü98], S. 45

Treffermengen reduziert werden. Es ist außerdem wünschenswert, anhand bestehender Zitatensammlungen ein Profil durch den Agenten erstellen zu lassen.

Manuelle Literaturrecherche

Neben den soeben dargestellten automatisch durch Literaturdienste oder Agenten durchgeführten Literaturrecherchen, werden von Gruppenmitgliedern täglich manuelle Anfragen an große Literaturdatenbanken, wie z.B. MEDLINE, gestellt. Diese Suche erfolgt über das *World-Wide Web*. Interessante Ergebnisse sollen auf einfache Art und Weise mit Hilfe der Maus (Kopieren/Einfügen) direkt aus dem Browser des Gruppenmitglieds in den lokalen Literaturdatenbestand übernommen werden können.

Erneut muß ein Übersetzer die erforderliche Konvertierung der Inhalte aus der Sprache der HTML-Repräsentation in die Sprache der lokalen Literaturobjekte durchführen.

Direkter Import aus Quellen

Es werden automatisiert Informationen aus der Nukleinsäuredatenbank *EMBL Nucleotide Sequence Database* in den globalen Datenraum eingearbeitet. Diese Einheiten enthalten neben Sequenzdaten und biologischen Annotationen ebenfalls Literaturstellen. Diese sollen voll automatisiert extrahiert und in die Literaturverwaltung übernommen werden. Auch hier wird ein entsprechender Übersetzer benötigt.

Neu erzeugte Literaturstellen

Im Rahmen des Sequenzierungsprojekts der Pflanze *A. thaliana* werden DNS-Abschnitte, die einen entsprechend hohen Stand der Annotation besitzen, bei der Nukleinsäuredatenbank *EMBL Nucleotide Sequence Database* eingereicht (*direct submission*). Bestandteil dieser Einreichung ist neben der Sequenzinformation und der gesamten biologischen Annotation auch die Information, wer an der Bestimmung der Sequenz und der Annotation beteiligt war. Dabei handelt es sich um neue Quellenangaben, die in die Literaturverwaltung aufgenommen werden müssen.

3.4 Anforderungen an ein CSCW-System

Im letzten Abschnitt wurden aus den dargestellten Gruppen in der Gen- und Genomanalyse Teilbereiche herausgegriffen und in Alltagssituationen dargestellt, für die ein CSCW-System entwickelt und prototypisch implementiert wurde. Anhand

dieser Szenarien wird im folgenden ein Anforderungskatalog für ein entsprechendes Groupwaresystem erstellt. Zunächst werden allgemeine Anforderungen herausgearbeitet, ehe auf gruppenspezifische Charakteristika eingegangen wird. Existierende Ansätze, die in der Literatur beschrieben wurden, werden vor dem Hintergrund dieses Katalogs im nächsten Abschnitt diskutiert (siehe 3.5).

3.4.1 Allgemeine Anforderungen

Zu Beginn werden allgemeine Anforderungen formuliert, die grundsätzlich für die Entwicklung aller Teilbereiche des Groupwaresystems gelten.

Objektmodell

Die im Rahmen der gruppeninternen Arbeiten anfallenden Informationen sollen als Objekte modelliert und, sofern erforderlich, persistent abgelegt werden. Der Einsatz objektorientierter Verfahren bietet sich aufgrund der hohen Komplexität der in diesem molekularbiologischen Anwendungsgebiet auftretenden Sachverhalte an. Der Hauptvorteil der Objektorientierung liegt in der leichten Erweiterbarkeit des Objektmodells (siehe z.B. [Boo96], [RBP+93]). Das Wissen in der molekularbiologischen Forschung nimmt ständig zu. Es entstehen daher immer neue Anforderungen an statische Informationsrepräsentationen und dynamische Interaktionen, die geeignet in bestehende Modelle und Systeme integriert werden müssen. Verfahren der Objekttechnologie (wie z.B. Vererbung, Polymorphismus) ermöglichen die notwendige Flexibilität.

Neben dieser Offenheit ist zusätzlich auf Skalierbarkeit zu achten. Das System muß an höhere Anforderungen angepaßt werden können, ohne daß existierende Komponenten dadurch negativ beeinflußt werden.

In dieser Arbeit erfolgt die Modellierung und Umsetzung unter Einsatz der Objekttechnologie.

Definition 7 (Objekt, nach [BS98])
Ein Objekt repräsentiert eine individuelle, identifizierbare Einheit, die abstrakt vorhanden ist und eine wohldefinierte Rolle im Problembereich spielt.

Es werden im folgenden grundsätzlich zwei Arten von Objekten unterschieden:

Definition 8 (Anwendungsobjekt)
Anwendungsobjekte repräsentieren Informationen des Anwendungsfeldes.

In dieser Arbeit ist das Anwendungsfeld die molekularbiologische Sequenzdatenanlyse. Anwendungsobjekte repräsentieren biologische Informationen.

Definition 9 (Administratives Objekt)
Administrative Objekte sind alle nicht-Anwendungsobjekte des Systems, die zur Aufrechterhaltung der Funktionalität erforderlich sind.

Ein Anwendungsobjekt ist beispielsweise die Repräsentation einer Proteinsequenz inklusive aller biologischer Zusatzinformationen. Ein Beispiel für ein administratives Objekt ist die Managementeinheit im System, das die Zugriffskontrolle auf Anwendungsobjekte durchführt.
Objekte werden durch Klassen beschrieben:

Definition 10 (Klasse, nach [RBP+93])
Eine Klasse beschreibt eine Gruppe von Objekten mit ähnlichen Eigenschaften (Attributen), gemeinsamem Verhalten (Methoden), gemeinsamen Relationen zu anderen Objekten und einer gemeinsamen Semantik.

Der im Zusammenhang mit Objekttechnologie oft erwähnte Vorteil der Wiederverwendbarkeit ist dabei nicht limitiert auf die Wiederverwendbarkeit von Teilen der Implementierung. Es existieren weitere Ebenen, auf denen Wiederverwendbarkeit möglich sein soll ([Nö98]): Wiederverwendbarkeit

- des Problemmodells,
- der Vorgehensweise zur Lösung des Problems,
- von Teilen der Architektur.

Einsatz eines Datenbankmanagementsystems

Zur persistenten Sicherung von Objekten soll ein Datenbankmanagementsystem (DBMS) zum Einsatz kommen. Die Funktionalität solcher kommerziell verfügbarer Managementsysteme soll vom Groupwaresystem benutzt werden.

Bei der Entwicklung der Systemarchitektur muß darauf geachtet werden, ab einem gewissen Abstraktionsniveau Unabhängigkeit vom konkret eingesetzten DBMS zu erhalten. Dadurch ist ein einfacher Austausch des DBMS möglich, ohne existierende, in der Abstraktion höherliegende Objekte modifizieren zu müssen.

Die Modellierung der Anwendungsobjekte, die unter Einsatz des DBMS persistent abgelegt werden sollen, bzw. ihrer zugrundeliegenden Klassen, muß datenbankunabhängig erfolgen. Lediglich administrative Objekte müssen auf einer bestimmten Ebene der Architektur DBMS-spezifisch realisiert werden. Diese Objekte realisieren die Schnittstelle zum eingesetzten DBMS.

Zugriffskontrollen unter Beachtung der Awareness-Problematik

Für Anwendungsobjekte müssen Zugriffskontrollen realisiert werden, um Konsistenz der enthaltenen Informationen vor dem Hintergrund des konkurrierenden Zugriffs sicherstellen zu können. Je nach Anwendung sind dabei geeignete Strategien (optimistisch oder pessimistisch) umzusetzen. Die dazu notwendigen Kontrollinstanzen sind ebenfalls als Objekte zu modellieren. Beim Entwurf des Systems muß der Awareness Aspekt beachtet werden. Wird einem Gruppenmitglied der Zugriff auf ein Anwendungsobjekt verweigert, da bereits ein anderes Gruppenmitglied oder ein Prozeß dieses Objekt bearbeiten, muß das Gruppenmitglied über die Gründe der Nicht-Verfügbarkeit informiert werden. Neben Informationen, von wem gewünschte Objekte seit wann bearbeitet werden, ist eine zusätzliche Kommunikationsunterstützung anzubieten. Dazu zählen die Angabe der Telefonnummern der Mitglieder sowie die Information, ob sie derzeit im System angemeldet sind. Die Möglichkeit, enstprechenden Personen auf einfache Art und Weise Emails schicken zu können, muß ebenfalls unterstützt werden. Dadurch wird die Kommunikationsarbeit zwischen den Gruppenmitgliedern erleichtert.

Während der Bearbeitung muß es einem Gruppenmitglied ermöglicht werden, die Verantwortung für ein Anwendungsobjekt auf ein anderes Gruppenmitglied übertragen zu können (z.B. aufgrund speziellen Fachwissens). Dazu muß das replizierte Objekt zwischen privaten Informationsräumen unter Wahrung der Konsistenz migrieren können. Das neu für das Objekt verantwortliche Gruppenmitglied kann nach Beendigung der Bearbeitungen das Objekt wieder in den gemeinsamen Gruppeninformationsraum einbringen.

Flexible Kommunikationsschicht

Das System wird in einem heterogenen Umfeld eingesetzt. Heterogenität existiert auf Plattformebene, d.h. unterschiedliche Rechnersysteme (*workstations*, PCs) unter verschiedenen Betriebssystemen (Unix, WindowsNT, OpenVMS, etc.), sowie auf der Ebene eingesetzter Programmiersprachen (C++, Java, perl). Es kann vorausgesetzt werden, daß alle Rechner des Systems über ein Netzwerk miteinander verbunden sind. Zur Kommunikationsunterstützung zwischen Objekten, die innerhalb dieses verteilten Systems auf unterschiedlichen Rechnern lokalisiert und in verschiedenen Programmiersprachen realisiert sein können, muß ein generischer Kommunikationsmechanismus zum Einsatz kommen. Der entfernte Prozeduraufruf (*RPC*) ist für diese Anforderung eine etablierte Technik, die, verallgemeinert und standardisiert, in der CORBA-Technologie angewendet wird (siehe z.B. [Bak97]).

Flexible und offene Systemarchitektur

Zur Reduzierung der Komplexität des Gesamtsystems wird die erforderliche Funktionalität als eine Menge individueller Komponenten abstrahiert, die miteinander in Relation stehen und über direkte oder indirekte Kanäle kommunizieren können. Jede Komponente erfüllt eine definierte Funktionalität, die über festgelegte Schnittstellen von anderen Komponenten genutzt werden kann. Komponenten können dabei Teile des biologischen Datenbestandes, aber auch graphische Bedienoberflächen sowie Management- bzw. Administrationseinheiten realisieren. Die Realisierung der jeweiligen Komponente, d.h. die zur Erfüllung der Anforderungen notwendigen Objekte, bleibt nach außen hin verdeckt (*encapsulation*). Lediglich Schnittstellenobjekte bieten ihre öffentlichen Methoden als Dienste an, die von Objekten weiterer Komponenten genutzt werden können. Die Kommunikation zwischen Komponenten, die sich vertikal und horizontal auf gleichen oder verschiedenen Abstraktionsebenen befinden können, wird durch die Kommunikationsschicht realisiert.

Durch diese flexible Modellierung und offene Architektur wird das Einfügen neuer Komponenten in das bereits bestehende System erleichtert. Von den bereits im System existierenden Komponenten müssen nur diejenigen modifiziert werden, die Dienste neuer Komponenten nutzen möchten. Es ist von den jeweiligen Komponenten und der neuen Funktionalität abhängig, wie umfangreich Modifikationen ausfallen können. Das Funktionieren alter Komponenten ist auch während der Anpassung an neue Rahmenbedingungen gewährleistet. Komponenten, die neue Funktionalitäten nicht nutzen, können unangetastet weiter betrieben werden. Werden Komponenten-interne Änderungen ohne Beeinflussung der Schnittstellenobjekte durchgeführt, kann dies ohne Änderungen externer Objekte transparent realisiert werden.

Je nach Komplexität einer Komponente ist intern eine hierarchische Architektur empfehlenswert, die verschiedene Abstraktionsebenen realisiert.

Automatisierter Datenimport

Generell werden Informationen aus externen Quellen in den organisationsweiten Informationsraum importiert. Dabei kann es sich um Sequenzdaten, Analyseergebnisse oder auch Literaturzitate handeln. Diese Quellen unterscheiden sich in ihren jeweiligen Informationsrepräsentationen. Um einen automatisierten Datenimport realisieren zu können, müssen Informationen aus der externen Darstellung extrahiert sowie in entsprechende gruppeninterne Objektrepräsentationen konvertiert werden. Dazu sind quellenspezifische Teilkomponenten notwendig (z.B. Parser, die externe Formate syntaktisch analysieren können).

Da der automatisierte Datenimport in allen Projekten aller Gruppen reali-

siert werden muß, wird eine allgemeine Komponente gefordert, die diese Aufgabe übernimmt. Lediglich quellenspezifische Teile dieser Komponente sollen für konkrete Anwendungen umgesetzt werden müssen. Die verbleibenden Teile sollen aufgrund ihres generischen Entwurfs wiederverwendet werden können. Die Komponente muß schließlich in die Systemarchitektur integriert werden können. In einem existierenden System befinden sich eine Vielzahl von Instanzen dieser Komponente, die sich nur in den Teilen, die die Informationsquelle betreffen, unterscheiden.

3.4.2 Spezielle Anforderungen

Für spezielle Anforderungen einzelner Gruppen kann nach dem vorangeganenen Abschnitt als Grundlage vorausgesetzt werden:

- Flexible plattform- und programmiersprachenunabhängige Kommunikationsschicht;
- Offene Architektur (Komponentenmodell);
- DBMS inklusive Zugriffskontrollen.

Im folgenden werden anhand alltäglicher Szenarien der jeweiligen Gruppen konkrete Anforderungen an das CSCW-System zusammengefaßt. Diese Anforderungen wurden beim Entwurf des Systems beachtet (siehe Kapitel 4.8.4) und umgesetzt (siehe Kapitel 5.5.3).

Wartung von Proteinsequenzen

Die Wartung von Proteinsequenzen betrifft eine bestimmte Gruppe bei MIPS. Folgende Anforderungen können abgeleitet werden, um das Gruppenziel einer redundanzfreien, vollständigen Proteinsequenzdatenbank hoher Qualität erreichen zu können:

- Einfügen neuer Proteinsequenzen in den gemeinsamen Informationsraum inklusive indirekter Kommunikationsunterstützung zwischen Gruppenmitgliedern:
  - interaktiv (individuelle Sequenzen aus verschiedenen Quellen);
  - semi-automatisiert (systematisch aus externen Quellen);
  - vollautomatisiert (systematisch aus Sequenzierungsprojekten) inklusive Unterstützung der Gruppenmitglieder im entsprechenden Sequenzierungsprojekt;

- Pflege existierender Proteinsequenzen (Modifikationen, Zusammenfassen von Objekten, etc.) unter Sicherstellung der Konsistenz;
- Erzeugen von Gruppenbewußtsein (Awareness);
- Private Informationsräume für Gruppenmitglieder mit der Möglichkeit der Objektmigration zwischen privaten Informationsräumen;
- Unterstützung von Telearbeit unter Wahrung der Konsistenz;
- Zugriff von entfernten, den Betreibern nicht bekannten Clients auf biologische Informationen unter Einsatz des Industriestandards *CORBA*;
- Präsentation biologischer Informationen der wissenschaftlichen Öffentlichkeit unter Einsatz verschiedener Techniken und Protokolle (WWW, *ftp*, CD-ROM).

Literaturverwaltung

Die Literaturverwaltung stellt einen zentralen projektübergreifenden Dienst dar, der von allen Gruppenmitgliedern aller Gruppen bei MIPS genutzt werden kann (siehe 3.3.3). Folgende Anforderungen können abgeleitet werden:

- Zentrale, konsistente Verwaltung von Literaturstellen;
- Gruppenmitglieder aller bei MIPS existierenden Gruppen können Literaturdienst nutzen;
- Transparente Verwaltung verschiedener Typen von Zitaten;
- Redundanzfreier Datenbestand;
- Verwaltung von Journalnamen;
- Lokale Ablage von Zusammenfassungen bei wissenschaftlichen Artikeln;
- Möglichkeit der umfassenden Recherche im lokalen, fokussierten Datenbestand;
- Automatisierter Import von Ergebnissen externer Literaturrecherchen;
- Agentenbasierte Literaturrecherche anhand individueller Profile von Gruppenmitgliedern;
- Individuelle Literatursammlungen für Gruppenmitglieder zur Unterstützung spezieller Teilgebiete innerhalb des Gruppenkontextes;

- Einfache Übernahme von Literaturstellen aus individuellen Literatursammlungen in globale Literaturdatenbank.

Diese Aufstellungen beschreiben ein Groupwaresystem, das Gruppen zum Erreichen des Gruppenziels unterstützt. Neben allgemeinen Teilen enthält dieses System gruppenspezifische Anwendungen. Im folgenden Abschnitt werden Groupwareansätze, die in der Literatur beschrieben wurden, vorgestellt und vor dem soeben aufgezeigten Hintergrund diskutiert. Es werden dabei Systeme betrachtet, die im Anwendungsgebiet Gen- und Genomanalyse angesiedelt sind.

3.5 Existierende Ansätze

Nachdem im vorangegangenen Abschnitt Anforderungen an ein Groupwaresystem für ausgewählte Aspekte in der Gen- und Genomanalyse formuliert wurden, werden für dieses Anwendungsgebiet existierende Lösungsansätze aufgezeigt und diskutiert.

3.5.1 Systeme zur molekularen Sequenzdatenanalyse

Die folgenden in der Literatur beschriebenen Systeme werden dabei erstmals unter dem Gesichtspunkt CSCW betrachtet:

- *Genome Annotation and Information Analysis (GAIA)* [BFS+98]
- *GeneQuiz* [ABL+99]
- *Genotator* [Har97]
- *Imagene* [MRDV99]
- *Multipurpose Automated Genome Project Investigation Environment (MAGPIE)* [GS96b], [GS96a]
- *Protein Extraction, Description, and ANalysis Tool (PEDANT)* [FM97a], [FM97b]
- *System for Easy Analysis of Lots of Sequences (SEALS)* [WK97]

Diese im Bereich der Gen- und Genomanalyse angesiedelten Systeme befassen sich mit der automatisierten Annotation einzelner Sequenzen bzw. ganzer Genome unter Einsatz etablierter Analysealgorithmen und -werkzeuge. Diese Programme sind geeignet, anhand vorliegender Sequenzen Analyseergebnisse zu erzeugen, die persistent verwaltet werden. Einige wichtige Vertreter solcher Analysealgorithmen sind:

- Sequenzvergleich
  - FASTA [Pea90]
  - BLAST [AGM+90]
- Multiples Sequenzalignment
  - CLUSTAL W [THG94]
- Strukturvorhersage
  - PREDATOR [FA97]
  - STRIDE [FA95]
  - COILS [LDS91]
- Proteineigenschaften
  - TMAP [PA94]
  - ALOM2 [KKD84]
  - SEG [WF93]
  - PROSEARCH [KLS92]
  - BLIMPS [WH92]

Bis auf *MAGPIE*, müssen bei allen Systemen Analyseprogramme lokal vorhanden sein. Bei *MAGPIE* werden auch entfernte Analysedienste herangezogen. Dazu werden Parameter an den entfernten Dienst übermittelt und die erhaltenen Ergebnisse in den lokalen Datensatz integriert. Der Datenaustausch erfolgt über Email.

Der Zugriff auf große externe Datensammlungen, wie Nukleinsäure- oder Proteinsequenzdatenbanken, erfolgt bei allen Systemen transparent. Es wurden entweder eigene Parser für die jeweiligen Formate entwickelt, um externe Informationseinheiten integrieren zu können, oder es wird auf Systeme zurüchgegriffen, die einen optimierten Zugriff erlauben (Recherchesysteme, wie z.B. *SRS* ([EA93])).

Das *Sequence Retrieval System (SRS)* wurde entwickelt, um auf Inhalte großer Textdateien Anfragen stellen zu können, die effizient beantwortet werden können. Hintergrund ist die Tatsache, daß alle großen molekularbiologischen Datenbanken als Textdateien vertrieben werden. *SRS* erzeugt zu jeder Textdatei eine Reihe von Indexdateien, die bestimmte Teilinformationen der in den Textdateien enthaltenen Informationseinheiten indizieren (realisiert als balancierte AVL-Bäume). Diese Indexdateien werden nach Erstellung in einem nachgeschalteten Schritt komprimiert. Von Anwendern gestellte Anfragen können durch Zugriff auf diese komprimierten Indexdateien optimiert beantwortet werden. Neben den von

SRS unterstützten Formaten können neue Formate zur Indizierung integriert werden. Dazu enthält *SRS* die Beschreibungssprache *Icarus*, in der Parser für Formate von Textdateien formuliert werden. *SRS* besitzt eine graphische Bedienoberfläche (realisiert in HTML), die die Formulierung komplexer, über mehrere Textdateien auszuführende Anfragen ermöglicht. Ergebnisse werden ebenfalls in HTML dargestellt. *SRS* ist ein statisches System. Werden Änderungen an den indizierten Textdateien durchgeführt, müssen die Indexdateien neu berechnet werden.[18] Der Einsatz von *SRS* in einem Umfeld, in dem viele Änderungen stattfinden wie etwa bei der Pflege von Proteinsequenzen, schließt sich daher aus praktischen Gründen aus.

Wurden einige Systeme entwickelt, um einzelne oder einige wenige Sequenzen automatisiert zu analysieren (*GAIA*, *Genotator*, *Imagene*), so unterstützt die Mehrheit die Analyse kompletter Genome (*GeneQuiz*, *MAGPIE*, *PEDANT*, *SEALS*).

Ergebnisse durchgeführter Analysen müssen zusätzlich zu den Sequenzdaten verwaltet werden. Überraschenderweise erfolgt bei der überwiegenden Mehrheit dieser Systeme die Verwaltung der biolgischen Informationen durch strukturierte Textdateien, die im Verzeichnisbaum des Dateisystems organisiert abgelegt werden (*GeneQuiz*, *Genotator*, *MAGPIE*, *SEALS*). *GAIA* und *PEDANT* verwenden relationale Datenbankmanagementsysteme (Sybase bzw. MySQL respektive).[19]

Neben einfachen und automatisiert durchführbaren Analysen sowie der Ablage daraus erzielter Resultate, ist die Präsentation von ermittelten Ergebnissen für Anwender von großer Bedeutung. Bis auf *SEALS*, das rein textbasiert ist, bieten alle Systeme graphische Oberflächen an. Hauptsächlich wird HTML, oftmals kombiniert mit Java Applets verwendet (*GeneQuiz*, *MAGPIE*, *GAIA*, *PEDANT*). *Genotator* und *Imagene* realisieren eigenentwickelte Oberflächen. Durch Verwendung der hypertextbasierten Querverweise können Abhängigkeiten zwischen biologischen Einheiten sehr leicht dargestellt und von Anwendern durch einfachen Mausklick verfolgt werden. Ein Wissen über die interne Organisation der Daten ist dabei nicht notwendig. Unabhängig von automatisierten Analysesystemen werden zunehmend generische graphische Bedienoberflächen entwickelt, die die Erfordernisse in diesem Anwendungsfeld realisieren sollen (siehe z.B. [HLLR98]). Diese Pakete, oftmals unter Einsatz der plattformunabhängigen Programmiersprache Java, erlauben die Integration existierender, hierarchisch tiefer angesiedelter Analyseprogramme bzw. Datenbanken.

[18] Die inkrementelle Aktualisierung von Indexdateien ist in *SRS* zwar vorgesehen, in der Praxis jedoch noch nicht einsetzbar.

[19] *Imagene* modelliert biologische Informationen als Objekte. Die persistente Organisation dieser Objekte ist in [MRDV99] jedoch nicht beschrieben.

Werden diese Systeme vor dem Hintergrund der rechnergestützten Gruppenarbeit betrachtet, muß festgestellt werden, daß kein derzeit in der Literatur beschriebenes System eine Gruppe von Anwendern unterstützt.

Teilweise wird einzelnen Anwendern die Möglichkeit angeboten, persönliche Annotationen einzufügen (*Genotator*, *GAIA*, *Imagene*, *MAGPIE*, *PEDANT*). Diese interaktiven Änderungen der Anwender werden jedoch unabhängig voneinander durchgeführt. Es existiert keine Nebenläufigkeitskontrolle. Da in den meisten Systemen keine Datenbankmanagementsysteme eingesetzt werden, ist die Sicherstellung der Konsistenz nicht gewährleistet. Vielmehr werden biologische Inhalte als Textdateien im Verzeichnisbaum abgelegt. Konkurrierende Zugriffe können sehr leicht zu Inkonsistenzen bzw. der Nichtverfügbarkeit ganzer Datensammlungen führen, wenn auf Mechanismen des Dateisystems zurückgegriffen wird.

Unabhängig, ob ein DBMS eingesetzt wird oder nicht, bedeutet das Fehlen einer Nebenläufigkeitskontrolle, daß bei gleichzeitiger Modifikation einer Informationseinheit durch zwei unabhängig voneinander agierende Anwender nach Abschluß dieser Modifikationen die Version als aktuelle Version sichtbar ist, die als letzte in den Datensatz übernommen wurde. Das parallele Bearbeiten einer Informationseinheit wird nicht erkannt. Im Zuge einer Re-Analyse bereits annotierter Objekte werden u.U. aufwendige manuelle Korrekturen durch automatisierte Prozesse wieder überschrieben. Insgesamt führt dies zu keiner dauerhaften qualitativen Verbesserung der enthaltenen biologischen Informationen.

3.5.2 Diskussion

Die dargestellten Systeme erleichtern die Anwendung einzelner Programme zur systematischen Exploration von Sequenzdaten. Die erforderlichen Eingabeformate und die erzeugten Ausgabeformate werden von den Systemen vor den Anwendern transparent erzeugt bzw. analysiert. Große Datenmengen werden automatisiert analysiert und erzielte Ergebnisse werden benutzerfreundlich dargestellt. Insbesondere die Unterstützung in der Analyse kompletter Genome ist eine deutliche Erleicherung für Wissenschaftler im Bereich der Gen- und Genomanalyse (vgl. 3.2.4). Das System *PEDANT* etwa enthält eine ständig wachsende Menge komplett sequenzierter und analysierter Organismen (29 komplett und 27 teilweise sequenzierte Genome[20]), so daß Beziehungen zwischen Eigenschaften unterschiedlicher Organismen untersucht werden können. Dadurch können neue Erkenntnisse über die Evolution, über Funktionen von Proteinen sowie allgemein über die Organisation von Organismen abgeleitet werden.

In der Natur existieren Ausnahmen, die durch den Einsatz generischer Algorithmen nicht behandelt werden können. Spezialisierte Wissenschaftler sind daher

[20] Stand März 2000

nicht ersetzbar, sondern müssen durch ein Groupwaresystem in ihrer Arbeit unterstützt werden. Sie müssen insbesondere in die Lage versetzt werden, automatisiert durchgeführte Annotationen nicht nur analysieren, sondern vor allem auch korrigieren zu können. Nicht nur neue Sequenzen müssen bearbeitet werden. Die Menge bereits bekannter Informationen muß ständig an den aktuellen Wissensstand angepaßt werden, damit die Qualität der Informationen erhalten bleibt bzw. verbessert wird. Da speziell in systematischen Genomsequenzierungsprojekten dies nicht von einzelnen Personen durchgeführt werden kann, ist die Gruppenunterstützung in der molekularbiologischen Sequenzdatenanalyse zwingend erforderlich. Eine Gruppenunterstützung wird jedoch bisher von keinem System angeboten.

Diese Arbeit stellt daher einen wichtigen Beitrag für die qualitative und quantitative Verbesserung der biologischen Informationen dar, die durch Gen- und Genomanalysen erzielt werden. Schwerpunkt in der Gruppenunterstützung ist nicht die transparente Integration externer Analysewerkzeuge und Datenbanken oder die Entwicklung spezialisierter graphischer Bedienoberflächen,[21] sondern die Unterstützung manueller Nachbearbeitungen sowie der systematischen Pflege erzielter Informationen durch eine Gruppe von Wissenschaftlern. Transparenter Datenbankzugriff und benutzerfreundliche graphische Bedienoberfläche müssen jedoch im Rahmen des Groupwaresysteme ebenfalls umgesetzt werden, wenn es die zu realisierende Unterstützung erfordert.

Das organisierte Ansammeln von Hypothesen und Wissen sowie deren ständige Erweiterung, Überprüfung und Anpassung ist eine entscheidende Grundlage für den biotechnologischen Forschungsbereich. Durch diese Arbeit kann die bestehende Situation des Informationsmanagements im Bereich der Gen- und Genomanalyse entscheidend verbessert werden.

[21] Dieses Problem wird von den beschriebenen Systemen bereits gelöst.

Kapitel 4

Entwurf eines Groupwaresystems für die Gen- und Genomanalyse

In diesem Kapitel wird gemäß den Anforderungen des vorangegangenen Kapitels ein Groupwaresystem entworfen. Nach dem Entwurf allgemeiner Systemkomponenten werden gruppenspezifische Lösungen modelliert. Auf implementierungsspezifische Aspekte wird im nächsten Kapitel eingegangen.

4.1 Datenhaltung

Das Groupwaresystem existiert als verteiltes System in einer heterogenen Umgebung. Unter einem verteilten System wird hier folgendes verstanden:

- Das System besteht aus einer Menge von Rechnern unterschiedlicher Plattformen.
- Die Rechner des Systems können über Kommunikationskanäle miteinander kommunizieren. Dabei existieren Festnetzverbindungen, aber auch Netzwerkverbindungen, die nur vorübergehend zur Verfügung stehen (z.B. ISDN-Verbindungen zur Unterstützung von Heimarbeit).
- Auf Rechnern des Systems ablaufende Softwareprozesse realisieren in ihrer Gesamtheit die Funktionalität des Groupwaresystems.
- Jedes Gruppenmitglied verfügt über einen eigenen Rechner zur Erledigung der geforderten Aufgaben abhängig vom Gruppenkontext (Koordination, gemeinsame Ressourcennutzung, individuelle Aufgaben, etc.).

Verteilte Systeme zeichnen sich durch besondere Charakteristika, speziell bei den Fehlermöglichkeiten aus, die beim Entwurf beachtet werden müssen (siehe 4.1.2).

4.1.1 Verteilte Datenhaltung

Da in allen Bereichen der Gen- und Genomanalyse große Mengen biologischer Daten verwaltet werden müssen, ist die eingesetzte Strategie der Datenhaltung von zentraler Bedeutung. Bevor eine Architektur zur Datenhaltung entworfen werden kann, müssen allgemeine Anforderungen zur Datenhaltung in einem verteilten System formuliert werden. Nach [Koc96] existieren folgende Grundforderungen:

- *Verfügbarkeit:* Ein Datenzugriff sollte jederzeit und von jedem Rechner aus für alle Gruppenmitglieder möglich sein.
- *Transparenz:* Sowohl Gruppenmitglieder als auch Anwendungsprogrammierer müssen nicht wissen, auf welchen Rechnern gewünschte Daten lokalisiert sind.
- *Konsistenz:* Trotz Nebenläufigkeit und Verteilung dürfen nur konsistente Datenzugriffe erfolgen.
- *Fehlertoleranz:* Die bisher genannten Forderungen sollten trotz im System auftretender Fehler nicht beeinträchtigt werden.

Bei den genannten, sich teilweise widersprechenden Forderungen sind Schwerpunkte zu setzen.[1] Insbesondere ist der Ort, an dem Daten abgelegt werden, geeignet zu wählen. Allgemein existieren zwei Extreme bei der Datenhaltung:

- *Zentrale* Datenhaltung: alle Daten werden an einem Ort verwaltet; Server bieten Zugriffsdienste an.
- *Replizierte* Datenhaltung: Im Netz existieren mehrere Kopien von Daten (Replikate), die lokal verwaltet werden. Es existiert kein Original. Konsistenz wird durch Kommunikation und Abstimmung der Zugriffe zwischen den Replikaten erreicht.

Neben diesen beiden Extremen existiert als Zwischenform die *hybride* Datenhaltung ([Koc96]). Bei dieser Variante existieren Replikate im Netz. Der schreibende Zugriff wird jedoch von einer zentralen Kontrollinstanz oder einem ausgezeichneten Replikat synchronisiert.

Der zentrale Ansatz hat den Vorteil, daß Konsistenz leichter sichergestellt werden kann, als wenn eine Vielzahl von Replikaten im Netz existieren. Der Nachteil liegt darin, daß diese Zentrale einen Flaschenhals darstellen und somit eine geringe Verfügbarkeit der Daten zur Folge haben kann. Z.B. sind bei einem Ausfall

[1] Hohe Verfügbarkeit bedeutet z.B. oftmals Abstriche bei der Konsistenz ([Koc96], S. 72).

des zentralen Rechners die dort verwalteten Daten nicht mehr zugänglich. Der replizierte Ansatz realisiert im Vergleich dazu eine höhere Verfügbarkeit und eine bessere Fehlertoleranz. Durch parallele Zugriffe kann die Performanz signifikant verbessert werden; das System läßt sich gut skalieren. Allerdings ist in diesem Fall die Sicherstellung der Konsistenz nur durch aufwendige Verfahren zu gewährleisten (siehe etwa [Bor91]). Bevor entschieden werden kann, welche Variante für das hier beschriebene System zum Einsatz kommen soll, müssen die Fehlermöglichkeiten analysiert werden.

4.1.2 Fehlermöglichkeiten in verteilten Systemen

Wie bereits erwähnt, wird durch die Gesamtheit der auf den Rechnern des verteilten Systems ablaufenden Softwareprozesse die Funktionalität des Groupwaresystems realisiert. In verteilten Systemen können einzelne Rechner oder Kommunikationskomponenten unabhängig voneinander jederzeit ausfallen. Je nach Organisation des Netzwerks kann der Ausfall einer Netzwerkkomponente zu Partitionierungen des Rechnernetzes führen.[2] In der Praxis kann nicht unterschieden werden, ob ein Rechner ausgefallen ist oder eine Partitionierung des Netzwerks vorliegt.

Es ist zu beachten, daß mobile Rechner oder Rechner, die bei der Ausübung von Telearbeit eingesetzt werden, über längere Zeiträume hinweg von allen Netzen und damit Kommunikationskanälen getrennt sein können. Diese Trennung ist jedoch absichtlich herbeigeführt. Damit diese Situation von Rechner- oder Netzwerkkomponentenausfällen unterschieden werden kann, ist die explizite Abmeldung des Rechners bzw. des entsprechenden Gruppenmitglieds durchführbar.

Neben dem Ausfall ganzer Rechner ist der Ausfall einzelner Softwarekomponenten zu behandeln, die auf einem Rechner des Netzwerks laufen. Die Nichtverfügbarkeit eines Dienstes bei gleichzeitiger Verfügbarkeit des entsprechenden Rechners muß automatisiert behandelt werden. Kontrollinstanzen erkennen solche Fehlerfälle und reagieren, indem der Dienst wieder zur Verfügung gestellt wird. Kann der Dienst auf einem bestimmten Rechner nicht automatisch wieder aktiviert werden, muß die Diensterbringung auf einen anderen Rechner innerhalb des lokalen Netzwerks ausweichen können. Clients, die diesen Dienst nutzen, müssen zum Zeitpunkt der gestellten Anfrage an den aktuellen Rechner verwiesen werden. Dazu sind entsprechende Vermittlungsdienste zur Verfügung zu stellen. Ist das automatisierte Bereitstellen eines Dienstes nicht möglich, muß eine zustän-

[2] Definition Partitionierung: „Ein Rechnernetz heißt partitioniert, wenn es zwei oder mehrere disjunkte Rechnermengen gibt, für die gilt: Kein Rechner aus einer der disjunkten Rechnermengen kann mit einem Rechner aus einer der anderen Rechnermengen kommunizieren. Ursachen der Partitionierung sind ausgefallene Kommunikationskomponenten, die eine Kommunikation zwischen den Partitionen verhindern.“ ([BS98], S. 194f)

dige Person sofort über dieses Problem informiert werden (z.B. via Email durch eine entsprechende Kontrollinstanz).

4.1.3 Eingesetzte Strategie der verteilten Datenhaltung

In dem hier beschriebenen Anwendungsfeld werden Daten in großen Mengen erzeugt. Diese Informationseinheiten müssen verwaltet und gepflegt werden. Dazu sind automatisierte Verfahren entwickelt worden, die eine manuelle Pflege nicht vollständig ersetzen, sondern lediglich unterstützen können. Vor diesem Hintergrund ist die Hauptforderung an die Datenhaltung die Sicherstellung der Konsistenz.

Das Groupwaresystem wird innerhalb einer Organisation in einem lokalen Netzwerk eingesetzt. Aufgrund der Stabilität eines lokalen Netzwerks werden in Bezug auf Verfügbarkeit keine Anforderungen an das Datenhaltungssystem gestellt. Es wird die hohe Verfügbarkeit innerhalb eines lokalen Netzwerks vorausgesetzt.

Biologische Informationseinheiten werden von einer Menge von Datenbankkomponenten verwaltet. Jeder Datenbankkomponente ist ein Rechner zugeordnet. Zugriffsdienste auf diese Daten werden über Server angeboten, die entsprechende Schnittstellen exportieren. Datenbankkomponenten und ihre Verwaltungseinheiten werden über die Rechner des Netzwerks verteilt, so daß im Idealfall pro Rechner nur eine Datenbankkomponente existiert. Der Ausfall eines Rechners hat dadurch nur die Nichtverfügbarkeit eines Datenservers zur Folge. Es können jedoch auch pro Rechner mehrere Datenbankkomponenten parallel existieren. Somit wird die Migration eines Servers auf einen anderen Rechner unterstützt.

Daten werden zentral verwaltet. Allerdings müssen im Umfeld der Gen- und Genomanalyse lange Transaktionen, die mehrere Tage dauern können, angeboten werden. Es wird daher die Strategie der hybriden Datenhaltung eingesetzt. Es können zu einer Dateneinheit, die in der Datenbankkomponente verwaltet wird, Replikate im Netz oder auch außerhalb des Netzwerks, z.B. auf mobilen Rechnern, existieren. Diese Replikate müssen von der Zentrale angefordert werden. Zur Sicherstellung der Konsistenz zwischen allen Replikaten einer Dateneinheit muß eine Nebenläufigkeitskontrolle realisiert werden.

4.2 Nebenläufigkeitskontrolle

4.2.1 Allgemeine Betrachtung

Existieren mehrere Prozesse, die konkurrierend auf ein Datum zugreifen können, entstehen nebenläufige Zugriffe. Zur Sicherstellung der Datenkonsistenz ge-

meinsamer Ressourcen müssen diese konkurrierenden Zugriffe koordiniert werden (*Nebenläufigkeitskontrolle*).

Erfolgt der Zugriff auf eine einzige Kopie von mehreren Prozessen aus, muß die *interne* Konsistenz sichergestellt werden. Dies kann durch Realisierung der ACID[3] Transaktionsforderungen erreicht werden. Existieren Replikate im System, muß neben der internen auch auf die *gegenseitige* Konsistenz geachtet werden.

Die Nebenläufigkeitskontrolle läßt sich nach zwei Grundstrategien realisieren. Bei der *pessimistischen* Nebenläufigkeitskontrolle wird das gleichzeitige Schreiben auf verschiedene Replikate unterbunden.[4] Das pessimistische Verfahren stellt sicher, daß immer nur ein Schreibzugriff auf ein Datum durchgeführt wird. Nach Abschluß dieses Zugriffs wird ein neuer Schreibzugriff erlaubt. Die Bearbeitungen eines Datums werden seriell abgefertigt.

Bei der *optimistischen* Nebenläufigkeitskontrolle wird jederzeit ein Zugriff auf Replikate erlaubt. Es wird angenommen, daß keine Konflikte auftreten. Dadurch wird eine höhere Verfügbarkeit erreicht als bei der pessimistischen Variante. Nach Abschluß von schreibenden Zugriffen auf Replikate muß überprüft werden, ob Konflikte aufgetreten sind. In einem solchen Fall kann entweder die modifizierende Operation zurückgenommen, der Konflikt automatisiert behoben oder durch das Gruppenmitglied manuell aufgelöst werden. Zur Konflikterkennung müssen an definierten Synchronisationspunkten Zustände von Replikaten überprüft und ggf. behandelt werden.

4.2.2 Eingesetzte Strategien zur Nebenläufigkeitskontrolle

Wie in 4.1 beschrieben, kommt eine hybride verteilte Datenhaltung zum Einsatz. Es können somit Replikate im System existieren. Grundsätzlich ist lesender Zugriff jederzeit auf allen Replikaten erlaubt. Der Zugriff auf potentiell veraltete Replikate wird in diesem Umfeld der Gen- und Genomanalyse in Kauf genommen. Zur Koordination der schreibenden nebenläufigen Zugriffe werden sowohl die pessimistische als auch die optimistische Variante eingesetzt.

Pessimistische Variante

Gemäß dem hybriden Ansatz der Datenhaltung werden bei der pessimistischen Nebenläufigkeitskontrolle schreibende Zugriffe auf Daten durch eine zentrale Instanz koordiniert. Diese Instanz hat Zugriff auf eine Datenbankkomponente, in der das Original eines Datums persistent abgelegt ist. Soll ein Datum von einem

[3] Das Akronym ACID beinhaltet die Forderungen *A*tomarität, Konsistenz (*C*onsistency), *I*solation und *D*auerhaftigkeit (siehe z.B. [J.D88]).

[4] Je nach Anwendung kann auch der lesende Zugriff einbezogen werden.

Prozeß verändert werden, muß bei der Zentrale eine Bearbeitungskopie des persistenten Originals angefordert werden. Nach Abschluß der Modifikationen auf dieser Kopie wird diese neue Version des Datums wieder an die Zentrale übergeben. Das Original in der Datenbankkomponente wird entsprechend aktualisiert. In der Zwischenzeit können keine weiteren Bearbeitungskopien zur Modifikation angefordert werden. Lesende Zugriffe finden immer auf dem Original statt und sind jederzeit möglich. Schreibende Zugriffe werden dadurch seriell, lesende Zugriffe parallel abgearbeitet.

Optimistische Variante

Bei der optimistischen Variante können Replikate zur Modifikation jederzeit bei der entsprechenden Zentrale angefordert werden. Nach Abschluß der Modifikationen wird das Replikat wieder an die zentrale Instanz übergeben. Diese führt eine Konfliktanalyse mit der Version in der Datenbankkomponente durch. Haben sich keine Konflikte ergeben, werden durchgeführte Änderungen übernommen. Ansonsten wird je nach Art des Konflikts dieser entweder automatisiert behoben (wenn z.B. verschiedene Bereiche eines Objekts betroffen sind) oder dem Gruppenmitglied zur manuellen Bearbeitung angezeigt.

Diskussion beider Varianten

Welche Variante zum Einsatz kommt, ist abhängig von den jeweiligen Anforderungen der verwalteten Daten und ihrem Gruppenkontext. Die pessimistische Nebenläufigkeitskontrolle vermeidet die Entstehung von Inkonsistenzen. Die optimistische Variante versucht, Konsistenz zu erzeugen. Die pessimistische Variante reduziert die Verfügbarkeit der Daten, erfordert jedoch keine aufwendigen Konfliktanalyseoperationen. Ihr Einsatz empfiehlt sich dann, wenn die Wahrscheinlichkeit, daß zwei Prozesse das gleiche Datum gleichzeitig bearbeiten wollen, relativ gering ist, oder eine automatisierte Konfliktbehebung nicht oder nur in wenigen konkreten Fällen durchführbar ist.

Da in diesem System beide Verfahren zum Einsatz kommen, werden beide Strategien im Kontext des Groupwaresystems allgemein entworfen.

4.3 Datenverwaltung

4.3.1 Reduzierung der Komplexität

Nachdem in den vorangegangenen Abschnitten die hybride Datenhaltung als verteilte Datenhaltungsstrategie ausgewählt wurde, müssen die Anforderungen der

Datenverwaltung aus dem vorangegangenen Kapitel aus dieser eher technischen Sichtweise betrachtet werden.

Es werden im folgenden Anwendungsobjekte betrachtet, die im Rahmen dieses Groupwaresystems erzeugt, gesichert und gepflegt werden. Neben projektspezifischen Informationen existieren Entitäten, die über Projektgrenzen hinweg Gültigkeit besitzen, d.h. auch nach Abschluß eines Projekts wertvolle Informationen beinhalten, die als Basis verfügbar sein sollen. Zur Vermeidung von Redundanz werden solche gemeinsamen Entitäten zentral verwaltet. Betrachtet man biologische Informationseinheiten generell, fällt die hohe Komplexität der Daten auf, insbesondere die Abhängigkeiten zwischen Informationseinheiten eines, aber auch unterschiedlicher Projekte. Man kann allgemein sagen, daß jedes Projekt in der Gen- und Genomanalyse neue Erkenntnisse und Informationen für diesen Forschungsbereich beisteuert, die von allen anderen Teilgebieten und damit auch Projekten genutzt werden können. Daher ist eine Vernetzung dieser vielschichtigen und in ihrer Struktur unterschiedlichen Informationen zu realisieren.

Aus der Sicht der Datenverwaltung muß diese Komplexität reduziert werden. Die Gesamtheit des Systems wird daher nicht in einem globalen Objektmodell abgebildet. Vielmehr werden Teilbereiche geschaffen, für die Objektmodelle entworfen werden. Objekte eines Objektmodells werden in assoziierten persistenten Speichern verwaltet. Relationen zwischen Objekten verschiedener Objektmodelle müssen modelliert werden. Als Ergebnis liegt ein hochgradig vernetztes System verwalteter Objekte vor.

Um den Vorzug der Wiederverwendbarkeit der Objekttechnologie nutzen zu können, werden zunächst Klassen extrahiert, die in allen oder möglichst allen Objektmodellen eingesetzt werden können. Diese bilden einen Basisbaukasten.

Neben diesen biologischen Anwendungsobjekten existieren eine Vielzahl weiterer Objekte, die technische Aspekte des Groupwaresystems realisieren. Diesen Objekten liegen abhängig vom konkreten Aufgabengebiet Objektmodelle zugrunde, die allgemeine und damit wiederverwendbare Teile enthalten.

4.3.2 Datenbankkomponente

Neben der Reduzierung der Komplexität der zu modellierenden anwendungsspezifischen sowie administrativen Objekte, ist auf höherer Abstraktionsebene die Reduzierung der Komplexität kommunizierender Einheiten des CSCW-Systems zu realisieren. Dies soll im folgenden unter dem Aspekt der Datenhaltung betrachtet werden.

Die Verwaltung von Objekten eines Objektmodells erfolgt unter Zuhilfenahme eines DBMS, d.h. Klasseninstanzen werden in persistenten Speichern abgelegt und verwaltet. Eine Instanz einer solchen Datenbank mit den für die Gruppenunterstützung notwendigen Zusatzfunktionalitäten wird als *Datenbankkomponente*

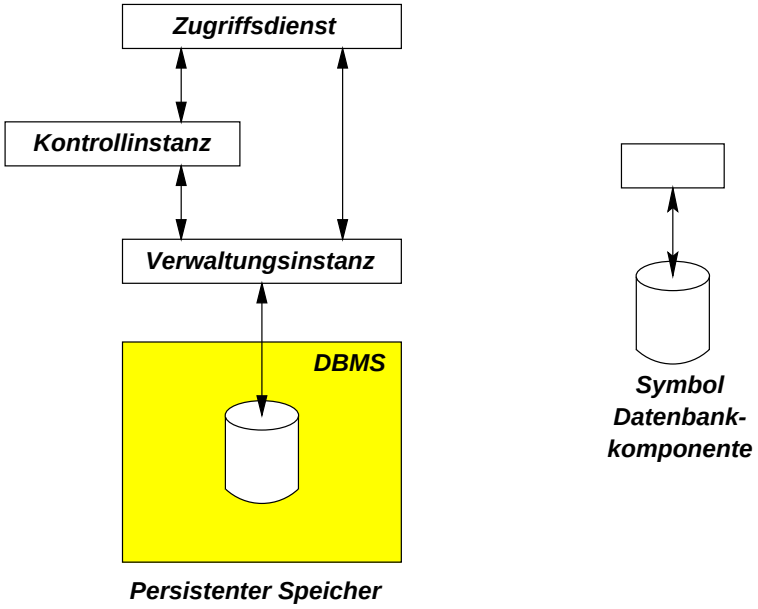

Abbildung 4.1: Datenbankkomponente bestehend aus den Einheiten *Persistenter Speicher* (verwaltet durch ein DBMS), *Verwaltungsinstanz, Kontrollinstanz* und *Zugriffsdienst* (in nachfolgenden Abbildungen wird eine Datenbankkomponente durch nebenstehendes Symbol dargestellt).

bezeichnet.

Definition 11 (Datenbankkomponente)
Eine Datenbankkomponente ist ein System bestehend aus folgenden Teilkomponenten:

- Persistenter Speicher*: eine oder mehrere Datenbanken, die von einem DBMS verwaltet werden.*
- Verwaltungsinstanz*: Instanz zur Verwaltung des persistenten Speichers.*
- Kontrollinstanz*: Instanz zur Realisierung der Nebenläufigkeitskontrolle sowie Gewährleistung von Redundanzfreiheit.*
- Zugriffsdienst*: realisiert lesenden und schreibenden Zugriff auf Objekte des persistenten Speichers.*

Ausgangspunkt für den Entwurf einer Datenbankkomponente ist ein Objektmodell, das einen Teilbereich des Anwendungsgebiets formal beschreibt. Objekte dieses Modells sind Anwendungsobjekte. Zusatzfunktionalitäten, die durch

Verwaltungsinstanz, Kontrollinstanz und Zugriffsdienst realisiert werden, werden ebenfalls durch Objekte, *administrative* Objekte, realisiert. Es existiert daher zu jeder Datenbankkomponente ein weiteres Objektmodell, durch das dieser administrative Teil modelliert wird. Daher wird zwischen dem *anwendungsspezifischen* und dem *administrativen* Objektmodell unterschieden. Während das anwendungsspezifische Objektmodell eigenständig existieren kann, ist das administrative Objektmodell vom anwendungsspezifischen Modell abhängig. So muß z.B. die Verwaltungsinstanz die Klassen des biologischen Modells kennen, um biologische Objekte persistent ablegen zu können. Der Übersichtlichkeit halber wird hier jedoch von zwei Objektmodellen gesprochen.

Anwendungsobjekte werden im persistenten Speicher der Datenbankkomponente abgelegt (siehe Abbildung 4.1). Die Verwaltungsinstanz stellt dazu die Basisfunktionen zur Verfügung (wie z.B. `insert`, `remove`, `get`, etc.). Darüberhinaus werden Indexdatenstrukturen von dieser Instanz verwaltet, um optimierte Zugriffe auf persistente Objekte durchführen zu können.

Die Kontrollinstanz stellt die Konsistenz der gespeicherten Objekte sicher. Je nach Anwendung wird entweder eine pessimistische oder optimistische Nebenläufigkeitskontrolle realisiert. Bei der pessimistischen Variante stellt die Kontrollinstanz sicher, daß ein Objekt nur genau einmal zur Modifikation angefordert wurde. Wird die optimistische Strategie verfolgt, erkennt die Kontrollinstanz Konflikte.

Der Zugriffsdienst schließlich realisiert die Schnittstelle der Datenbankkomponente zu externen Komponenten des CSCW-Systems. Diese Teile des Systems können nur über die angebotenen Dienste auf persistente Objekte zugreifen. Dadurch wird die interne Datenverwaltung inklusive der administrativen Instanzen nach außen hin verdeckt (transparenter Objektzugriff). Ein Client muß lediglich Dienste in Anspruch nehmen. Wissen über interne Datenverwaltung ist nicht notwendig. Insbesondere ist das eingesetzte DBMS verdeckt. Das Austauschen eines DBMS kann daher innerhalb der Datenbankkomponente durchgeführt werden, ohne daß weitere Teile des Systems davon betroffen wären.

In 3.2.1 wurden verschiedene Informationsräume definiert. Zur Realisierung dieser abstrakten Informationsräume werden Datenbankkomponenten eingesetzt. Ein Informationsraum besteht dabei aus einer einzelnen oder einem Verbund von Datenbankkomponenten (siehe Abbildung 4.2).

4.4 Datenzugriff

Bisher wurde in diesem Kapitel auf Datenhaltung und Datenverwaltung in verteilten Groupwaresystemen eingegangen. Wie im vorangegangenen Abschnitt beschrieben, wird der Zugriff auf persistente Objekte durch spezielle Zugriffsdienste

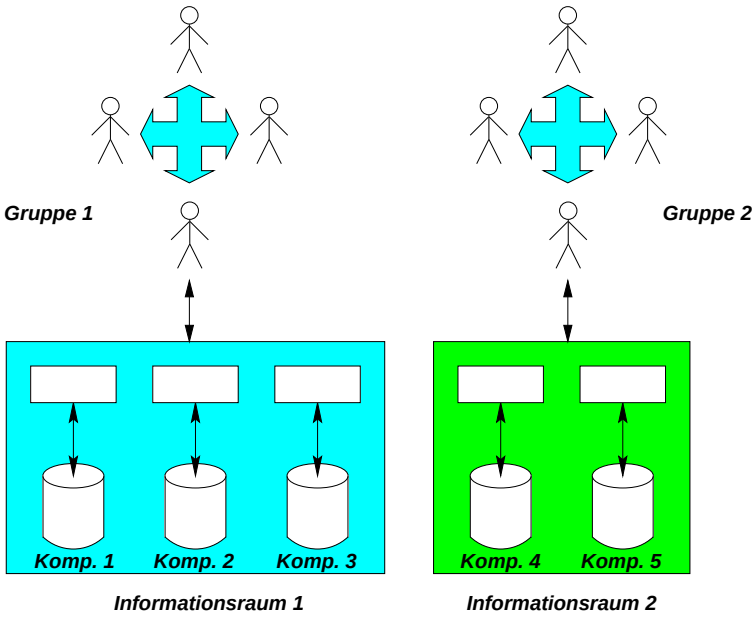

Abbildung 4.2: Ein Informationsraum I besteht aus $n \geq 1$ Datenbankkomponenten.

der jeweiligen Datenbankkomponenten ermöglicht. Im folgenden wird dargestellt, wie Dienste von Servern angeboten sowie von Clients in Anspruch genommen werden können. Dazu wird eine Kommunikationsschicht eingeführt.

4.4.1 Kommunikationsschicht

Aufgabe dieser Schicht ist die Realisierung einer flexiblen Client/Server-Kommunikation in einer heterogenen Umgebung: verschiedene Plattformen (PCs, *workstations*), unterschiedliche Programmiersprachen (z.B. Java, C++, perl) und mehrere Betriebssysteme (Windows, Unix). Ausgangssituation ist somit ein heterogenes verteiltes System, in dem Rechner über ein lokales Netzwerk (LAN) miteinander kommunizieren können.

Die Kommunikationsschicht muß eine Vielzahl unterschiedlicher Daten im Rahmen der Objektkommunikation transportieren. Neben Parametern und Ergebnissen eines Prozeduraufrufs werden Anforderungen, Awareness-Informationen oder auch Fehlermeldungen übertragen.

Diese Schicht muß außerdem unabhängig von konkreten Anwendungen die notwendige Funktionalität anbieten. Abgesehen von anwendungsspezifischen Teilen, die konkrete Schnittstellen und Parameter, etc. betreffen, kann die Kommunikationsschicht leicht in neue Anwendungen des Groupwaresystems integriert

werden. Bestehende Applikationen dürfen von der Integration neuer Anwendungen nicht betroffen sein.

Entfernter Prozeduraufruf

Aus der Sicht eines Clients müssen entfernte Dienste wie lokale Prozeduren aufrufbar sein. Ein Client soll weder über interne Datenverwaltungen noch über Dienstinanspruchnahmen noch über aktuelle Netzwerksituationen detailliert informiert werden müssen. Für dieses Problemfeld existiert als etablierte Technik der *Remote Procedure Call (RPC)* (siehe z.B. [BS98], [Sch92a], [Sch92b]). Dabei werden Daten durch Prozeduraufrufe zwischen Prozessen über einen Kommunikationskanal ausgetauscht. Die beteiligten Prozesse befinden sich in unterschiedlichen Adreßräumen. Beim *synchronen* RPC wartet der Aufrufer so lange, bis das Ergebnis des entfernten Prozeduraufrufs vorliegt. Im Gegensatz dazu wird der Aufruferprozeß beim *asynchronen* RPC nicht blockiert.

Aus Beschreibungen von Schnittstellen heraus, die in einer standardisierten Schnittstellenbeschreibungssprache abgefaßt werden (z.B. IDL bei CORBA), wird von Übersetzern Quellcode generiert, der die für die Kommunikation über ein Netzwerk notwendigen Routinen (*stubs*) realisiert. Dieser generierte Code wird zu den Client- und Server-Applikationen gebunden (siehe Abbildung 5.5).

Der RPC-Mechanismus wird nun auf die Objekttechnologie übertragen. Dies bedeutet, daß ein Client keine entfernte Prozedur aufrufen kann. Vielmehr entsteht eine Kommunikation zwischen einem lokalen Client-Objekt und einem entfernten Server-Objekt. Das Client-Objekt nutzt einen Service des Server-Objekts, d.h. ruft eine öffentliche Methode des entfernten Objekts auf.

Vermittlung

Um hohe Flexibilität zu erreichen, sind Dienste nicht an spezielle Rechner gebunden, sondern können innerhalb des Netzwerks migrieren. Beim Ausfall eines Rechners soll z.B. ein Dienst auf einem anderen Rechner innerhalb des Netzwerks zur Verfügung gestellt werden, um eine hohe Verfügbarkeit dieses Dienstes zu gewährleisten. Der Client ist nicht darüber informiert, auf welchem Rechner welcher Dienst angeboten wird (Ortstransparenz). Vielmehr sollen zur Laufzeit Anforderungen von Clients entsprechend aktueller Situationen an geeignete Server weitergereicht werden.

Für diese Zuordnung von Client und Server ist ein Vermittlungsdienst notwendig. Ein Server registriert seinen Ort sowie die zur Verfügung gestellten (exportierten) Schnittstellen bei einer Vermittlungseinheit. Diese Informationen werden Clients zur Verfügung gestellt.

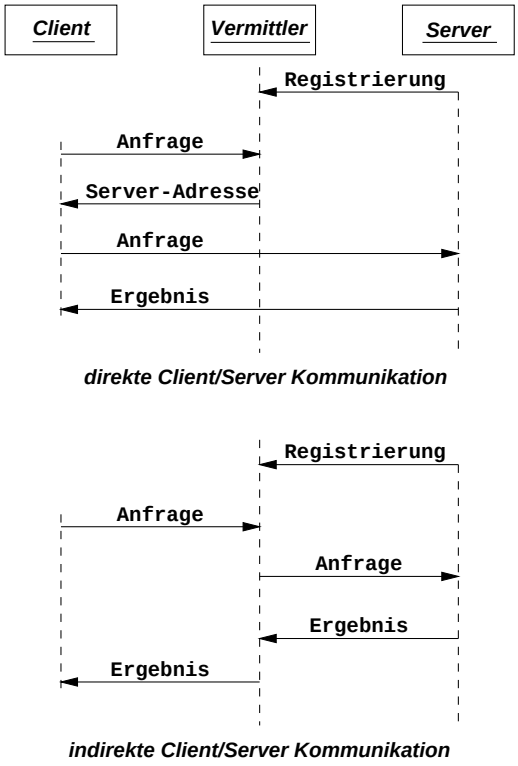

Abbildung 4.3: Die Zuordnung von Client und Server wird durch einen Vermittlungsdienst realisiert. Je nach Realisierung liegt *direkte* oder *indirekte* Client/Server-Kommunikation vor.

Auf Vermittler basierende Client/Server-Kommunikation kann direkt oder indirekt realisiert sein (siehe Abbildung 4.3). Bei der *direkten* Kommunikation senden Clients, nach Inanspruchnahme von Vermittlern, Daten direkt an Server. Bei der *indirekten* Variante wird die Kommunikation zwischen Client und Server durch den Vermittler realisiert. Clients stellen Anfragen an Vermittler. Diese wählen anhand der registrierten Schnittstellen geeignete Server aus, und leiten Client-Anforderungen an diese weiter. Ergebnisse werden von Vermittlern an die Clients zurückgeleitet.

In einem Netzwerk können eine Vielzahl von Vermittlern existieren. Verfügen diese Vermittler über ein gemeinsames Protokoll, ist eine Kommunikation zwischen ihnen möglich, um geeignete Server für Anfragen auswählen zu können. Dadurch ist innerhalb eines Netzwerks eine Lastverteilung möglich, wodurch Ressourcen besser genutzt werden können.

Kommunikationsprotokoll

Ist eine Verbindung zwischen Client und Server bzw. dem dazwischengeschalteten Vermittler zustande gekommen, sind die technischen Voraussetzungen erfüllt, um einen Datenaustausch zwischen den beteiligten Prozessen durchführen zu können. Damit diese Kommunikation geregelt ablaufen kann, ist ein Kommunikationsprotokoll notwendig. Dieses Protokoll muß sowohl auf Server- als auch auf Clientseite implementiert sein. Neben Parameterdaten eines entfernten Prozedur- bzw. Methodenaufrufs (Datentyp, Wert) müssen auch Daten, die den angeforderten Dienst spezifizieren, übertragen werden. Treten Fehler auf, werden Fehlermeldungen transportiert. Das Kommunikationsprotokoll unterstützt somit eine Vielzahl unterschiedlicher Datenarten.

Um einen Einsatz in einer heterogenen Umgebung zu ermöglichen, muß ein Kommunikationsprotokoll von konkreten Programmiersprachen, Betriebssystemen und Plattformen unabhängig sein. Das Protokoll wird daher auf einer höheren Abstraktionsebene entworfen. Anschließend muß die plattformspezifische Realisierung durchgeführt werden. Dabei ist zu beachten, daß Plattformspezifika (wie z.B. unterschiedliche Präsentationen von Datentypen auf verschiedenen Plattformen) auf die einheitliche Darstellung gemäß dem allgemeinen Kommunikationsprotokoll abgebildet werden müssen. Die dazu notwendigen Befehlsabfolgen können von Übersetzern plattform- und programmiersprachenspezifisch automatisiert erzeugt werden.

4.4.2 Der Standard *CORBA*

Ein allgemeiner Mechanismus des Methodenaufrufs entfernter Objekte, der die soeben genannten Anforderungen (Vermittlung, Kommunikationsprotokoll, etc.)

erfüllt, ist der Standard *CORBA (Common Object Request Broker Architecture)* der *OMG (Object Management Group)* (siehe z.B. [Bak97], [SGR99]). CORBA ist lediglich eine Spezifikation. Die Implementierung wird von Firmen realisiert, die ihre Produkte auf dem Markt anbieten. Alternativ werden von akademischen Gruppen CORBA-Implementierungen angeboten, die oftmals im nichtkommerziellen Umfeld kostenlos eingesetzt werden dürfen.

CORBA verbindet zwei wichtige Trends in der derzeitigen Softwareindustrie: die objektorientierte Softwareentwicklung sowie die Realisierung flexibler Client/Server Applikationen. CORBA erweitert den RPC-Mechanismus, indem verschiedene Abstraktionsebenen definiert werden: der CORBA-interne Vermittler (ORB) abstrahiert die komplexe Netzwerkprogrammierung, CORBA *services* bieten auf Systemebene Funktionalitäten an, und CORBA *facilities* sind standardisierte Ansätze, Lösungen für Probleme in speziellen Anwendungsgebieten anzubieten. Zwar wird durch diese Abstraktionen die Realisierung der netzwerkweiten Kommunikation in einer verteilten Anwendung erleichtert, die inhärente Komplexität der Anwendungsapplikation bleibt jedoch erhalten.

CORBA bietet die für die Kommunikationsschicht notwendige Funktionalität an, da es die Forderungen nach Plattform- und Programmiersprachenunabhängigkeit erfüllt. Außerdem wird das Paradigma der Objektorientierung unterstützt, nach dem in einem System nur kommunizierende Objekte existieren. Die CORBA Technologie kann daher in der Kommunikationsschicht dieses Groupwaresystems eingesetzt werden.

4.5 Infrastruktur des Groupwaresystems

In den vorangegangenen Abschnitten wurden eine Datenhaltungsschicht und eine Kommunikationsschicht eingeführt. Werden Anwendungsobjekte in der Datenhaltungsschicht unter Sicherstellung der Konsistenz verwaltet, realisiert die Kommunikationsschicht einen allgemeinen Kommunikationskanal in einem heterogenen verteilten System.

Diese Schichten sind hierarchisch angeordnet (siehe Abbildung 4.4). Die Basis bildet die Datenhaltungsschicht, in der Datenbankkomponenten angesiedelt sind. Zugriffe auf Dienste der Datenhaltungsschicht werden durch die Kommunikationsschicht ermöglicht. Über der Kommunikationsschicht ist schließlich die Anwendungsschicht angesiedelt. Diese Schicht enthält alle Arten von Anwendungen, die z.B. Dienste von Datenbankkomponenten nutzen. Kommunikation zwischen Applikationen der Anwendungsschicht erfolgt ebenfalls unter Einsatz der Kommunikationsschicht. Dadurch wird sowohl vertikale als auch horizontale Kommunikation unterstützt.

Die allgemeine Architektur, bestehend aus Datenhaltungsschicht und Kom-

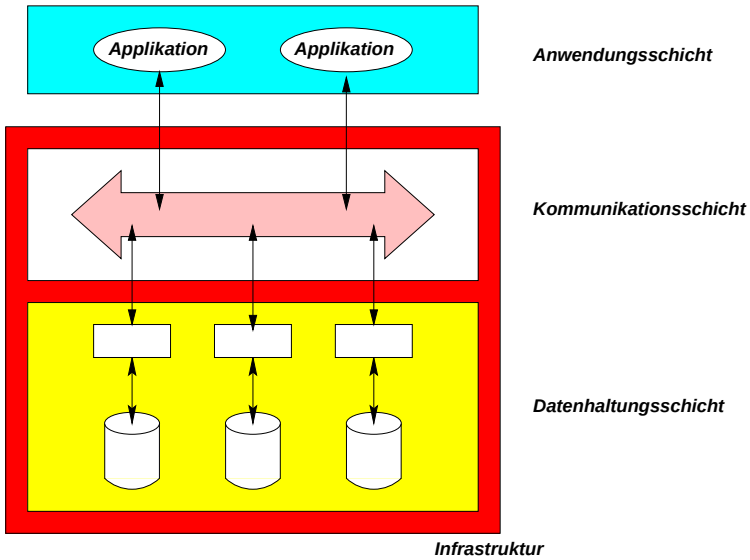

Abbildung 4.4: Hierarchische Anordnung der unterschiedlichen Schichten des generischen CSCW-Systems.

munikationsschicht, bildet die *Infrastruktur* des hier beschriebenen Groupwaresystems. Diese Infrastruktur wird von allen Applikationen des CSCW-Systems genutzt, da sie die elementaren Basisdienste anbietet: die persistente Verwaltung von Objekten unter Sicherstellung der Konsistenz sowie die flexible Kommunikation in einem verteilten heterogenen System. In Kapitel 5.5.3 wird auf Implementierungsdetails dieser Infrastruktur eingegangen.

4.6 Automatisierter Datenimport

Basierend auf der soeben dargestellten Infrastruktur soll eine allgemeine Komponente entworfen werden, die einen automatisierten Datenimport aus externen Quellen ermöglicht. Ein automatisierter Datenimport erfordert das Extrahieren von Informationen aus externen Datenquellen sowie die Konvertierung in adäquate Repräsentationen im Kontext des Groupwaresystems.

Zur Realisierung dieser Funktionalität wird als Systembaustein der *Informationswandler* eingeführt (siehe Abbildung 4.5).

Definition 12 (Informationswandler)
Ein Informationswandler ist ein System bestehend aus folgenden Einheiten:

- *Importeinheit*

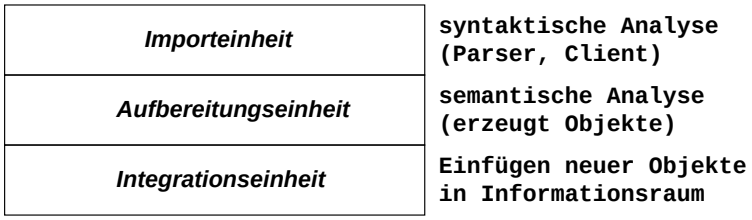

Abbildung 4.5: Der Systembaustein *Informationswandler* und seine geschichtete Architektur.

- *Aufbereitungseinheit*
- *Integrationseinheit*

Die *Importeinheit* realisiert den Zugriff auf Daten externer Quellen (siehe Abbildung 4.6). Eine Realisierung kann durch Parser erfolgen, die Formate externer Quellen syntaktisch verarbeiten können. Dies ist Voraussetzung für Zugriffe auf enthaltene Informationen. Zur Realisierung von Parsern müssen die Grammatiken der zugrundeliegenden Sprachen beschrieben werden.

Eine konkrete Importeinheit kann auch als Client realisiert sein, falls ein Zugriff auf externe Quellen via Client/Server-Kommunikation möglich ist (etwa unter Einsatz von CORBA). Proprietäre Zugriffsmöglichkeiten, die i.d.R. Bestandteil von Datenbankdistributionen sind, müssen entsprechend integriert werden. Unabhängig von der konkreten Implementierung bieten Importeinheiten extrahierte Informationen der nachgeschalteten Aufbereitungseinheit an.

Die *Aufbereitungseinheit* greift auf angebotene Daten zu. Import- und Aufbereitungseinheit verfügen dazu über eine standardisierte interne Repräsentation externer Daten, die unabhängig von der Sprache der Quelle ist. Die Aufbereitungseinheit extrahiert unter Verwendung spezifizierter Zugriffsschnittstellen entsprechende Informationen und erzeugt Objekte im Kontext des Groupwaresystems. Es erfolgt dadurch eine Datenkonvertierung aus externen Formaten in interne Repräsentationen der anwendungsspezifischen (hier biologischen) Informationseinheiten, die als Objekte dargestellt werden. Für diese Umsetzung sind komplexe semantische Methoden notwendig, die die Abbildung interner Repräsentationen von Informationseinheiten auf Objekte des Groupwaresystems realisieren. Diese semantischen Methoden müssen manuell für jedes Objektmodell des Groupware-Systems realisiert werden.

Das Einfügen neu erzeugter Objekte in entsprechende Informationsräume des Groupwaresystems ist Aufgabe der *Integrationseinheit*. Diese fungiert aus

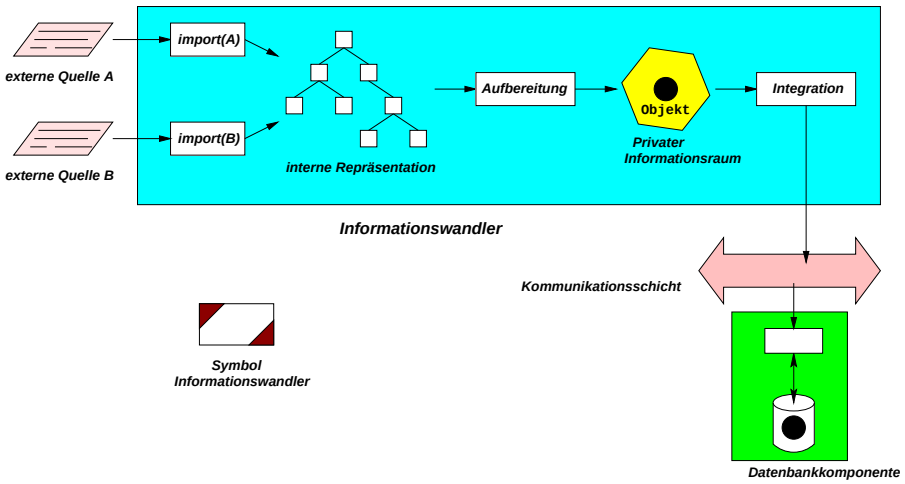

Abbildung 4.6: Detaillierte Darstellung der Einheiten eines Informationswandlers (in nachfolgenden Abbildungen wird ein Informationswandler durch links unten aufgeführtes Symbol dargestellt).

Sicht einer Datenbankkomponente als Client, der neue Objekte unter Anwendung exportierter Zugriffsdienste in entsprechende Informationsräume integriert. Die Realisierung der dazu notwendigen Kommunikation erfolgt durch die Kommunikationsschicht.

Während die Importeinheit spezifisch für jeweilige externe Quellen entwickelt werden muß, sind Aufbereitungs- und Integrationseinheiten davon unabhängig. Sie operieren mit festgelegten internen Strukturen sowie Anwendungsobjekten des Anwendungskontexts und kommunizieren mit administrativen Objekten des Groupwaresystems. Das bedeutet insbesondere, daß für jede Datenbankkomponente lediglich eine Aufbereitungs- und eine Integrationseinheit entworfen werden müssen, die beliebig von Informationswandlern wiederverwendet werden können.

Ein Informationswandler wird durch ein Gruppenmitglied oder einen Agenten aktiviert. Ein Agent kann etwa eine externe Quelle überwachen und neue Einträge erkennen, die in einen lokalen Informationsraum importiert werden sollen (siehe Abbildung 4.7). Gruppenmitglieder können interaktiv einen Import anstoßen. Im Rahmen des Datenimports erzeugte Objekte werden zunächst im persönlichen Informationsraum des aktivierenden Prozesses erzeugt und anschließend von der Integrationseinheit in den gemeinsamen Gruppen-Informationsraum migriert.[5]

Aufgrund der komplexen Semantik im molekularbiologischen Anwendungsfeld und oftmals nur unzureichend definierter Datenformate externer Datenbank-

[5]Sowohl Gruppenmitglieder, als auch Agenten verfügen über persönliche Informationsräume.

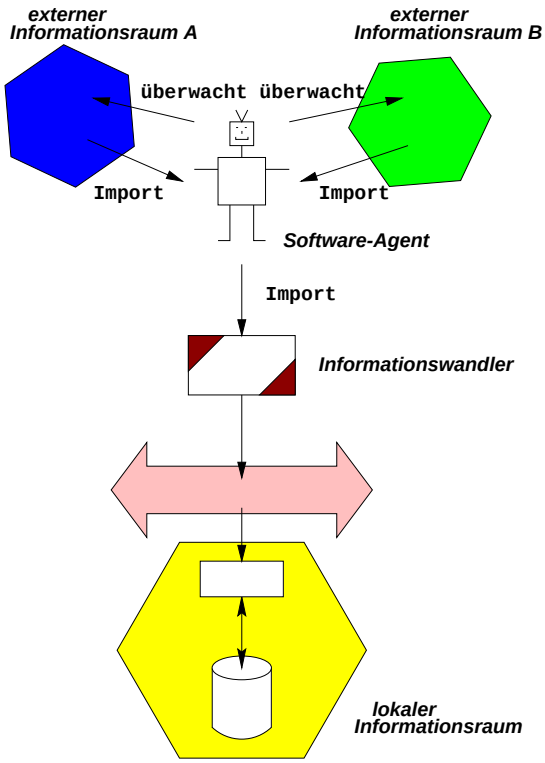

Abbildung 4.7: Ein Informationswandler kann durch einen Software-Agenten aktiviert werden, der z.B. externe Quellen überwacht und neue Informationen in den lokalen Informationsraum integriert.

betreiber, sind semantische Methoden in der Aufbereitungseinheit nicht immer umfassend genug realisierbar. Daher muß, abhängig von der Quelle, nach der automatisierten Aufbereitung ein manuelles Eingreifen durch Gruppenmitglieder ermöglicht werden, bevor das Objekt in den Gruppen-Informationsraum eingefügt werden kann.

Aufbereitungs- und Integrationseinheit werden dazu entkoppelt (siehe Abbildung 4.8). Von der Aufbereitungseinheit erzeugte Objekte sind zunächst in persönlichen Informationsräumen von Gruppenmitgliedern enthalten. Es können nun umfassende Modifikationen an den Objekten durchgeführt werden, ohne daß eine Nebenläufigkeitskontrolle notwendig ist: lediglich Eigentümer persönlicher Informationsräume besitzen Erlaubnis zum schreibenden Zugriff. Nach Abschluß manueller Bearbeitungen migriert die Integrationseinheit Objekte aus persönlichen in den gemeinsamen Informationsraum.

Sollen neue externe Quellen in das System integriert werden, muß ein entsprechender Informationswandler realisiert werden. Die Integration in das bestehende

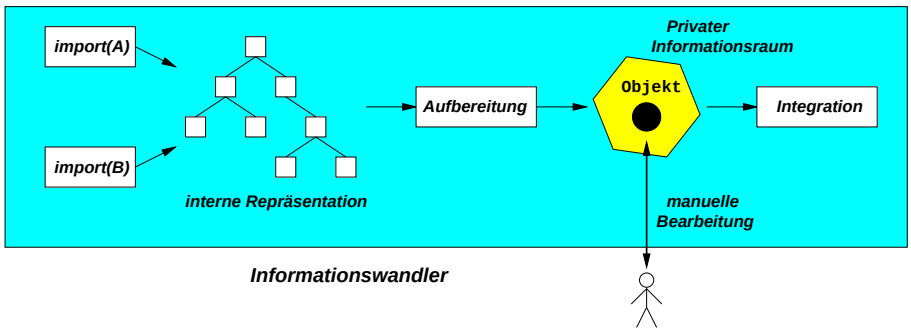

Abbildung 4.8: Entkoppelung von Aufbereitungs- und Integrationseinheit eines Informationswandlers, um manuelle Bearbeitungen zu ermöglichen.

System ist aufgrund der offenen Architektur inkrementell durchführbar. Soll zu bestehender Datenbankkomponente und existierenden Informationswandlern eine neue externe Quelle integriert werden, muß lediglich die quellenspezifische Importeinheit realisiert werden. Alle übrigen Komponenten können wiederverwendet werden.

4.7 Awareness

Wie bereits im letzten Kapitel angedeutet, findet Gruppenkommunikation auf Sach- und Koordinationsebene statt. Der Informationsaustausch die Sache betreffend erfolgt rechnergestützt (via Email) oder direkt (z.B. spontane Diskussionen zwischen Gruppenmitgliedern vor der Kaffeemaschine oder beim gemeinsamen Mittagessen). Ein strukturierter rechnergestützter Informationsaustausch ist auf Sachebene in diesem Anwendungsgebiet aufgrund der komplexen Semantik aufwendig zu realisieren und wird in dieser Arbeit nicht näher betrachtet.

Informationen zur Koordination der Arbeit in einer Gruppe, d.h. Informationen über Tätigkeiten anderer Gruppenmitglieder in der Vergangenheit, Gegenwart und Zukunft, müssen rechnergestützt ausgestauscht werden, insbesondere vor dem Hintergrund asynchroner Arbeit und Telearbeit. Durch diese Art Informationsaustausch wird „ein gemeinsames Verständnis der Gruppenmitglieder über die gemeinsamen Tätigkeiten und damit über die Gruppenarbeit gefördert.“ ([Bü98], S. 41). Awareness-Informationen erzeugen bei Gruppenmitgliedern ein psychologisches Existenzbewußtsein: Einzelpersonen sind Bestandteil einer Gruppe und arbeiten koordiniert auf das gemeinsame Gruppenziel hin.
Nach [Bü98] wird Awareness wie folgt verstanden:

Definition 13 (Awareness)
Awareness bezeichnet das Wissen, auf dem das Verständnis des Arbeitsgeschehens beruht.

Somit ist Awareness das Bindeglied (*glue*) zwischen Mitgliedern einer Gruppe. Dadurch wird eine effektivere Gruppenarbeit erreicht, da die Kommunikationsarbeit der Gruppenmitglieder unterstützt wird.

In dieser Arbeit betrifft Awareness die Inhalte gemeinsamer Informationsräume (*work space awareness*). Folgende Informationen sind für Gruppenmitglieder zur Koordination ihrer Arbeit interessant:

- *Gibt es neue Objekte im Informationsraum?*
- *Wurde ein bestimmtes Informationsobjekt modifiziert?*
- *Fehlen Objekte im Informationsraum?*
- *Wenn ich ein Informationsobjekt nicht bearbeiten kann: von welchem Gruppenmitglied wird dieses Objekt seit wann bearbeitet?*[6]

Diese Art von Anfragen können von Gruppenmitgliedern jederzeit gestellt werden. Anfragen wie „Wurde ein bestimmtes Objekt modifiziert?“ können einem Agenten übergeben werden, der Gruppenmitglieder sofort über derartige Ereignisse unterrichtet.[7]

Um diese Art von Anfragen beantworten zu können, müssen zu Objekten in Informationsräumen neben inhaltlichen Daten zusätzlich entsprechende Verwaltungsinformationen abgelegt werden. Interessant sind z.B. Zeitstempel, die beschreiben, wann ein Objekt kreiert oder modifiziert wurde. Welches Gruppenmitglied die entsprechende Operation ausgeführt hat, muß ebenfalls eingetragen werden.

Diese Informationen werden in einer Klassenhierarchie, den *Awareness-Klassen*, modelliert. Diese Hierarchie ist unabhängig von konkreten Anwendungen. Unterschiedliche Anforderungen an Awareness-Informationen werden durch Klassen innerhalb dieser Hierarchie abgebildet. Instanzen von Awareness-Klassen sind *Awareness-Objekte*.

Awareness-Daten sind für jedes Objekt im Informationsraum spezifisch. Es erfolgt daher eine Aggregation des Awareness-Objekts in das konkrete Objekt des Anwendungsfelds, wodurch eine explizite persistente Verwaltung von Awareness-Objekten vermieden wird (siehe Abbildung 4.9): die Verwaltungsinstanz einer Datenbankkomponente verwaltet insbesondere auch Awareness-Objekte. Allerdings

[6] Beim Einsatz einer pessimistischen Strategie zur Nebenläufigkeitskontrolle.
[7] In der Bioinformatik sprich man auch von *Alert-Systemen*.

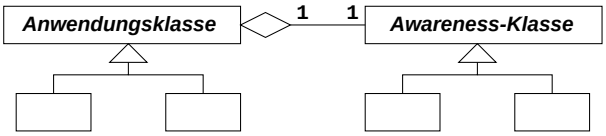

Abbildung 4.9: Aggregation von Awareness-Informationen in Anwendungsklasse.

muß die Verwaltungsinstanz entsprechende Anfragen auf Awareness-Objekte unterstützen; vom Anfragedienst müssen sie angeboten werden.

Awareness-Objekte werden wie Anwendungsobjekte durch die Kommunikationsschicht von Datenbankkomponenten zu entsprechenden Anwendungsapplikationen transportiert. Das bedeutet, daß zur Unterstützung des Awareness-Konzepts die Infrastruktur des Groupwaresystems eingesetzt werden kann. Awareness-Informationen werden allgemein modelliert sowie realisiert und können daher für alle CSCW-Applikationen wiederverwendet werden. Sollten zukünftige Anwendungen weitergehende Awareness-Informationen benötigen, können neue Awareness-Klassen durch Ableitung in die bestehende Hierarchie der Awareness-Klassen integriert werden. Bereits in Datenbanken existierende Awareness-Objekte sind von solchen Modifikationen der Klassenhierarchie nicht betroffen.

4.8 Ausgewählte Applikationen

Nachdem in den vorangegangenen Abschnitten eine allgemeine Infrastruktur für Applikationen des Groupwaresystems entworfen wurde, werden im verbleibenden Teil dieses Kapitels exemplarisch Applikationen dargestellt, die basierend auf der eingeführten Infrastruktur realisiert und in der Gruppe eingesetzt wurden.

4.8.1 Zentrale Literaturverwaltung

Ziel der zentralen Literaturverwaltung ist es, einen fokussierten und redundanzfreien Datenbestand an Literaturzitaten unterschiedlichen Typs zu unterhalten, der projektübergreifend von allen Gruppenmitgliedern genutzt werden kann (vgl. 3.3.3).

Gemäß der beschriebenen Architektur wird die zentrale Literaturverwaltung als Datenbankkomponente in der Datenhaltungsschicht modelliert. Gemäß der Definition einer Datenbankkomponente müssen folgende Teile entworfen werden:

- Persistenter Speicher, der Literaturzitate enthält;

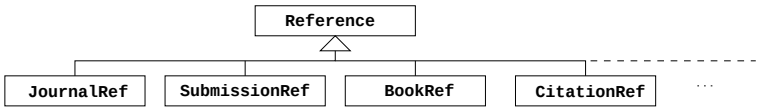

Abbildung 4.10: Literaturobjektmodell.

- Verwaltungsinstanz, die diesen persistenten Speicher verwaltet;
- Kontrollinstanz, die eine Strategie zur Nebenläufigkeitskontrolle realisiert und damit Konsistenz der Daten sicherstellt;
- Zugriffsdienst, über den die Datenbank angesprochen werden kann.

Persistenter Speicher

Als persistenter Speicher wird eine Datenbank angenommen, die von einem DBMS verwaltet wird. Unter Einsatz des DBMS müssen Literaturobjekte in der Datenbank abgelegt werden. Ein Literaturobjekt ist dabei eine Instanz einer Literaturklasse (siehe Abbildung 4.10). Folgende Literaturklassen werden benötigt: in einem wissenschaftlichen Journal veröffentlichter Artikel, *submission*,[8] Buch, *citation* sowie nicht publizierte Zitate (*unpublished*). Diese Klassen ergeben sich aus der Analyse der Literaturreferenzen, die für die Proteinsequenzdatenbank *PIR-International* definiert wurden. Die weiteren zu unterstützenden Projekte verfügen über keine darüberhinausgehenden Literaturklassen.

Diese verschiedenen Literaturklassen werden als Hierarchie modelliert. Die allen Klassen gemeinsamen Informationen, wie etwa Autorennamen, werden in einer abstrakten Basisklasse[9] `Reference`[10] abgelegt. Alle, konkrete Literaturtypen beschreibenden Klassen sind von dieser abstrakten Basisklasse abgeleitet. Sie enthalten entsprechende Klassenvariablen und Methoden. Sollte ein neuer Literaturtyp notwendig werden, kann eine weitere Klasse leicht in die Hierarchie integriert werden. In der Literaturdatenbank enthaltene Objekte sind von diesen Modifikationen des Objektmodells nicht betroffen.

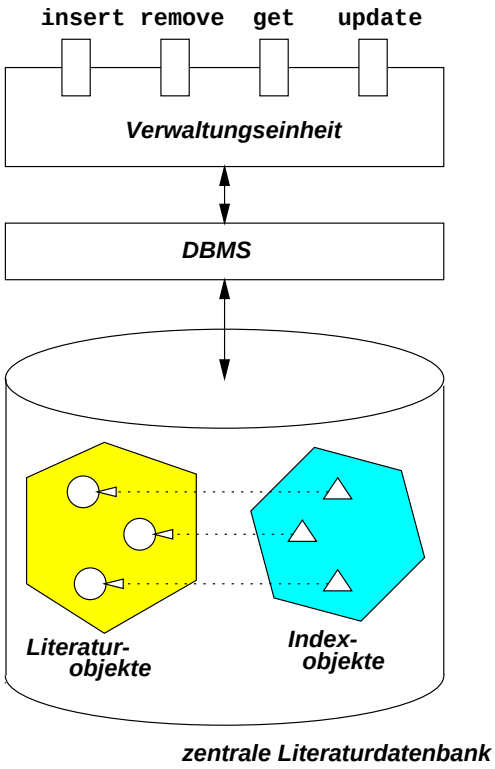

Abbildung 4.11: Verwaltungseinheit einer Datenbankkomponente.

Verwaltungseinheit

Die Verwaltungseinheit einer Datenbankkomponente verwaltet den zugehörigen persistenten Speicher, d.h. Objekte, die in den persistenten Speicher eingefügt wurden. Sie stellt Operationen zur Verfügung, die auf die Datenbank angewendet werden können: `insert`, `remove`, `get` und `update` (siehe Abbildung 4.11). Zusätzlich werden von dieser Instanz Index-Datenstrukturen verwaltet, die die effiziente Beantwortung von Anfragen ermöglichen. Diese Strukturen müssen in jeder den Datenbankzustand modifizierenden Transaktion aktualisiert werden. Die Verwaltungseinheit realisiert somit die Basisfunktionalität der Datenbank.

[8] Direkte Einreichung von Daten bei einer öffentlichen Datenbank (siehe auch 3.2.2).

[9] Abstrakte Basisklasse bedeutet, daß von dieser Klasse keine Instanz (Objekt) erzeugt werden kann.

[10] In der Bioinformatik hat sich der englische Begriff *reference* für Literaturverweise etabliert und wird daher auch in dieser Arbeit verwendet.

Kontrollinstanz

Basierend auf der Verwaltungsinstanz realisiert die Kontrollinstanz folgende Anforderungen:

- Nebenläufigkeitskontrolle zur Sicherstellung der Konsistenz der Daten;
- Redundanzfreiheit.

Der Zugriff auf Objekte der Datenbank ist nur über die von Verwaltungsinstanzen bereitgestellten Operationen möglich.

Optimistische Nebenläufigkeitskontrolle: Wie beschrieben, können optimistische und pessimistische Strategien zur Nebenläufigkeitskontrolle eingesetzt werden. I.d.R. werden Literaturobjekte, nachdem sie in den Datenbestand eingefügt wurden, nicht mehr modifiziert. Ausnahmen sind lediglich wissenschaftliche Artikel, zu denen evtl. zu einem späteren Zeitpunkt Zusatzinformationen, wie etwa die Zusammenfassung des Artikels, eingefügt werden. Aufgrund der geringen Anzahl durchgeführter Modifikationsoperationen und der damit einhergehenden geringen Wahrscheinlichkeit, daß zwei Gruppenmitglieder gleichzeitig dasselbe Objekt bearbeiten wollen (bei ca. 100.000 Literaturobjekten und 17 Gruppenmitgliedern), wird bei dieser Datenbankkomponente eine optimistische Strategie zur Nebenläufigkeitskontrolle eingesetzt (siehe Abbildung 4.12).

Wie bei der pessimistischen Variante muß ein Replikat eines Objekts zur Modifikation von der jeweiligen Anwendung bei der Datenbankkomponente angefordert werden (*checkout*). Diese Anforderung ist immer erfolgreich, da bei der optimistischen Variante angenommen wird, daß kein Konflikt auftreten wird. Nach Abschluß der Modifikationen wird das Replikat wieder an die Datenbankkomponente übergeben (*checkin*). Die Kontrollinstanz überprüft, ob ein Konflikt entstanden ist. Dazu werden das modifizierte Replikat sowie das persistente Objekt der Datenbank verglichen. Im Konfliktfall wird die Anwendung, die den *checkin*-Dienst angefordert hat, sofort darüber informiert. In einer interaktiven Anwendung können dem Gruppenmitglied beide Versionen des Objekts präsentiert werden: die aktuelle Version in der Datenbank sowie das modifizierte Replikat. Durch geeignete graphische Darstellungen können Unterschiede zwischen den Objekten ansprechend hervorgehoben werden. Die inhaltliche Entscheidung, welche Version des Literaturobjekts in der Datenbank gehalten werden soll, liegt in der Verantwortung des Gruppenmitglieds.

Tritt bei Ausführung eines *checkin*-Dienstes kein Konflikt auf, wird die persistente Version des Objekts durch das eingereichte Replikat ersetzt. Im assoziierten Awareness-Objekt werden entsprechende Informationen aktualisiert. So wird der Zeitstempel aktualisiert, der den Zeitpunkt der letzten Modifikation beschreibt.

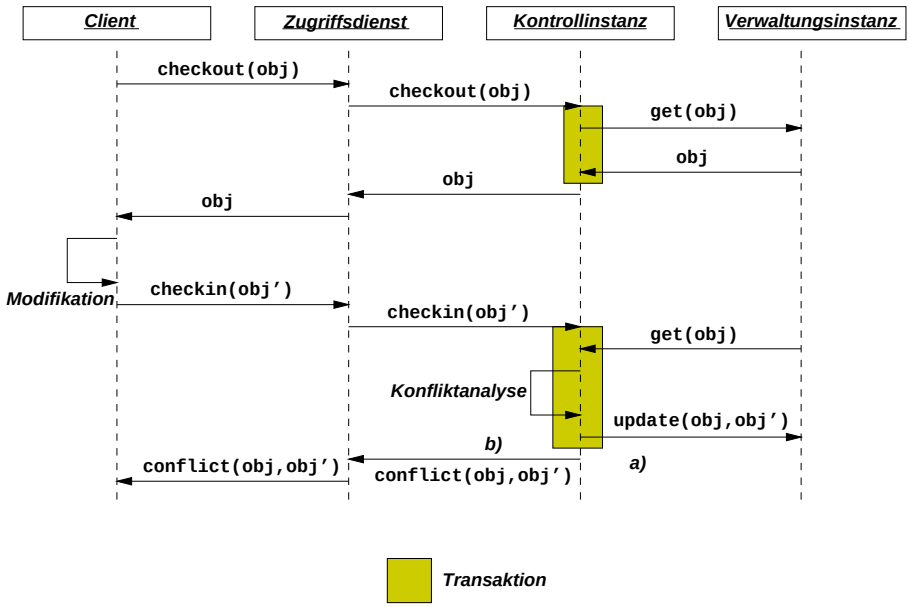

Abbildung 4.12: Optimistische Variante zur Nebenläufigkeitskontrolle. Im Fall *a)* ist kein Konflikt aufgetreten und das persistente Objekt kann durch das modifizierte Replikat ersetzt werden. Im Fall *b)* trat ein Konflikt auf. Beide Objekte werden an den Clientprozeß durchgereicht. Dort kann je nach Anwendung auf diese Situation eingegangen werden.

In Abbildung 4.13 ist der Konfliktfall dargestellt. Zwei Clients fordern das gleiche Objekt zur Modifikation an. Es werden transiente Replikate erzeugt, die u.a. den Zeitstempel der letzten Modifikation des persistenten Originals enthalten, und an die Clientprozesse übergeben. Nach Abschluß der Bearbeitungen erfolgt zunächst ein *checkin* vom Prozeß *Client 2*. Es tritt kein Konflikt auf, da das persistente Objekt in der Zwischenzeit nicht manipuliert wurde. Daher werden die durchgeführten Modifikationen in den persistenten Speicher übernommen. Der Zeitstempel der letzten Veränderung wird im persistenten Objekt aktualisiert. Zu einem späteren Zeitpunkt aktiviert der Prozeß *Client 1* ebenfalls den Dienst *checkin*. Bei der Konfliktanalyse wird aufgrund der unterschiedlichen Zeitstempel der letzten Veränderung zwischen persistentem Objekt und transientem Replikat ein Konflikt erkannt. Es wird der Konfliktbehandlungsdienst aktiviert.

Zur Konfliktbehandlung muß die Kontrollinstanz Wissen über die Semantik der möglichen Literaturobjekte besitzen. Die Kontrollinstanz ist ebenfalls als Klasse modelliert. Diese Klasse verfügt über Wissen die Klassenhierarchie der Literaturklassen betreffend, d.h. der Entwurf der Klasse, die instantiiert die Funktionalität der Kontrollinstanz realisiert, ist abhängig vom Anwendungsob-

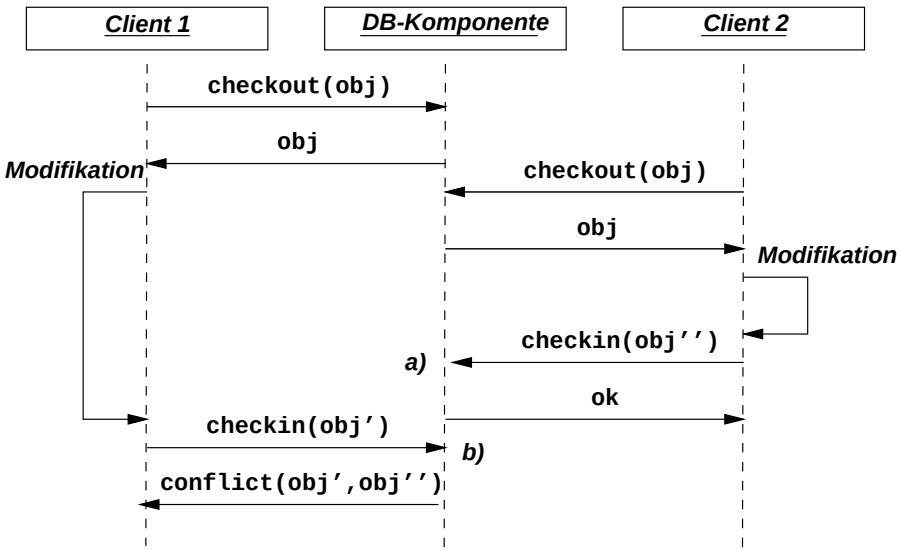

Abbildung 4.13: Konflikterkennung bei der optimistischen Variante der Nebenläufigkeitskontrolle. Im Fall *a)* kann der *checkin*-Dienst erfolgreich durchgeführt werden. Im Fall *b)* wurde in der Zwischenzeit das persistente Objekt modifiziert. Das *checkin* schlägt fehl.

jektmodell. Aufgrund dieser einseitigen Abhängigkeit können semantische Konflikterkennungsstrategien umgesetzt werden. Durch diese Trennung zwischen Anwendungs- und administrativem Objektmodell können Literaturklassen unabhängig von den eingesetzten Strategien zur Konsistenzsicherung und Gewährleistung der Redundanzfreiheit entworfen werden. Insbesondere können die eingesetzten Strategien während des Betriebs angepaßt bzw. verändert werden, ohne daß das Anwendungsobjektmodell davon betroffen wäre.

Redundanzfreiheit: Die Sicherstellung der Redundanzfreiheit liegt ebenfalls in der Kompetenz der Kontrollinstanz. Sie stellt sicher, daß Literaturstellen nur genau einmal in der Datenbank enthalten sind. Dazu ist Wissen über die Semantik des jeweiligen Literaturtyps notwendig. Wie oben beschrieben, ist der Klasse, die die Kontrollinstanz realisiert, die Klassenhierarchie der Literaturklassen bekannt.

Wie in 2.4.2 dargestellt, ist das Proteinsequenzdatenbankprojekt *PIR-International* eine Kollaboration von MIPS mit Partnern aus den USA und Japan. Für die Unterstützung der Datenbankgruppe bei MIPS ist lediglich der amerikanische Partner von Bedeutung. Der interne Datenabgleich zwischen der Gruppe MIPS und der amerikanischen Gruppe erfolgt im wöchentlichen Rhythmus. Diese recht lockere Koppelung der beteiligten Gruppen kann zu Redundanzen in den Datenbe-

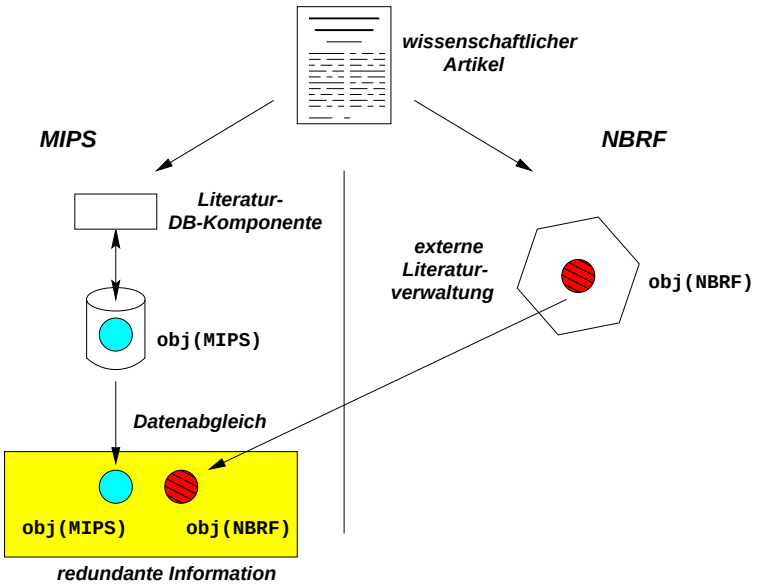

Abbildung 4.14: Entstehung redundanter Information im lokalen Informationsraum am Beispiel eines Literaturzitats im Rahmen des Proteinsequenzdatenbankverbunds *PIR-International*.

ständen führen, speziell bei Literaturzitaten, da auch Mitglieder weiterer Projekte bei MIPS diesen zentralen Literaturdienst nutzen.

Es kann der Fall eintreten, daß innerhalb einer Woche die gleiche Literaturstelle sowohl in Amerika, als auch bei MIPS erfaßt wird (siehe Abbildung 4.14). Jeder neuen Literaturstelle wird ein eindeutiger Objekt-Identifikator zugeteilt. Dadurch ist es etwa in einer HTML-Seite sehr leicht realisierbar, einen *link* auf das entsprechende Literaturobjekt in der Literaturverwaltung zu generieren. Ohne daß die Gruppe bei MIPS darüber informiert wurde, wird beim Datenabgleich ein Replikat (aus inhaltlicher Sicht) in den organisationsweiten Informationsraum eingeführt. Einer neuen Literaturstelle wurde in Amerika dabei ein anderer Identifikator zugeteilt als bei MIPS, da beide Gruppen über streng getrennte Identifikatorengenerierungssysteme verfügen.[11] Inhaltlich beschreiben diese beiden Literaturobjekte jedoch das gleiche Zitat. Im Rahmen des Datenabgleichs muß diese Situation erkannt werden, um eine doppelte Aufnahme des Zitats und damit Redundanz in der zentralen Literaturverwaltung zu vermeiden.

Verallgemeinert man dieses Szenario, so werden von lose gekoppelten Gruppen, die über lokale Replikate eines gemeinsamen Informationsraums verfügen, neue Objekte nebenläufig eingefügt. Aufgrund der losen Koppelung geschieht das

[11] Das Format der erzeugten Identifikatoren ist jedoch bei beiden Gruppen gleich.

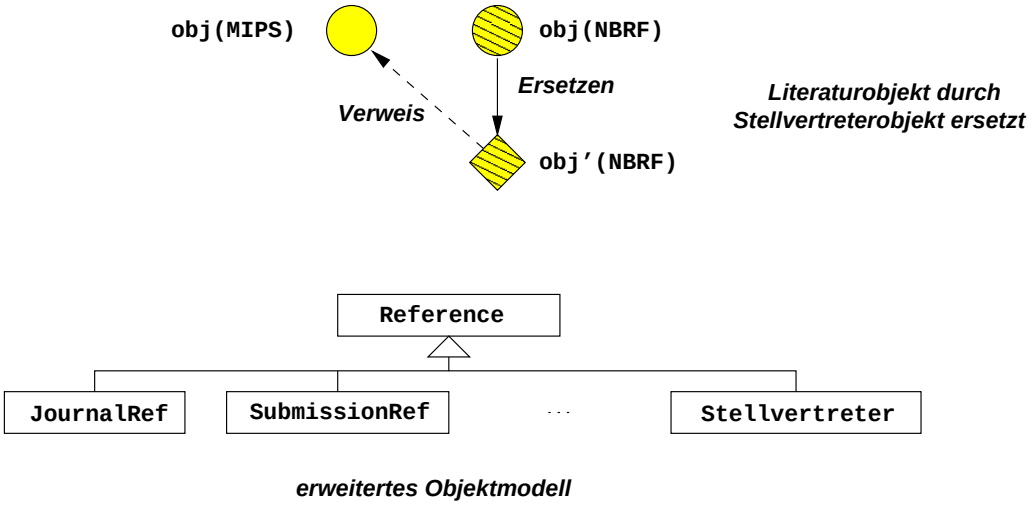

Abbildung 4.15: *Oben:* Ein redundantes Literaturobjekt (`obj(NBRF)`) wird durch ein Stellvertreterobjekt (`obj'(NBRF)`) ersetzt, das auf ein Literaturobjekt (`obj(MIPS)`)verweist. *Unten:* Zur Modellierung des Stellvertreterkonzepts wird das Objektmodell der Litertaturzitate um eine Stellvertreterklasse erweitert.

ohne Wissen der Partner. Im Rahmen des Datenabgleichs entstehen Redundanzen, die behandelt werden müssen.

Das Erkennen dieser Situation allein reicht nicht aus. Vielmehr muß adäquat darauf reagiert werden. Der Identifikator einer Literaturstelle wird u.a. auch in Annotationsdatenfeldern von Proteinsequenzdatenbankobjekten eingetragen. Eine Forderung ist, daß ein Identifikator immer zu einem gültigen Literaturobjekt in der zentralen Literaturverwaltung verweisen muß. Außerdem darf ein Identifikator nicht verändert werden. Es ist daher beim Datenabgleich nicht möglich, redundante Zitate, die sich lediglich in ihren Identifikatoren unterscheiden, zu ignorieren. Vielmehr müssen diese synonymen Identifikatoren, d.h. auf das gleiche Zitat verweisend, von der Literaturverwaltung behandelt werden.[12]

In dieser Situation werden Stellvertreterobjekte eingeführt (siehe Abbildung 4.15). Ein Stellvertreterobjekt verfügt wie ein Literaturobjekt über einen Identifikator. Der Inhalt dieses Objekts besteht jedoch lediglich aus dem *gültigen Identifikator* im Kontext der zentralen Literaturverwaltung. D.h. ein Stellvertreterobjekt enthält als Klassenvariable einen Identifikator, der auf ein tatsächliches Literaturobjekt verweist. Das Stellvertreterobjekt wird als pseudo-Literaturobjekt betrach-

[12] Diese Identifikatoren sind somit keine Schlüssel im Datenbankkontext.

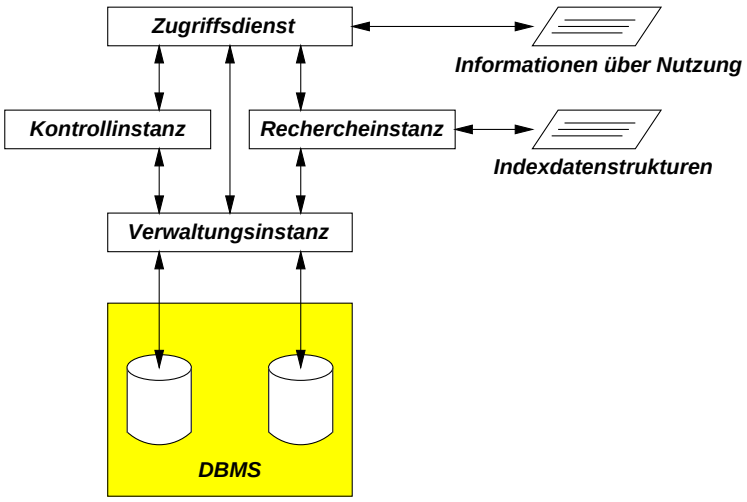

Abbildung 4.16: Der Zugriffsdienst realisiert die Schnittstelle nach außen und verdeckt dadurch interne Bereiche einer Datenbankkomponente und deren technische Umsetzung.

tet.

Im oben dargestellten Szenario wird bei MIPS ein neues Stellvertreterobjekt erzeugt. Dieses erhält als Identifikator den Identifikator des Literaturobjekts, das von der amerikanischen Gruppe generiert wurde. Die Klassenvariable wird auf den Wert des Identifikators des korrespondierenden Literaturobjekts bei MIPS gesetzt. Das Stellvertreterobjekt realisiert somit einen Querverweis auf ein Literaturobjekt innerhalb der Literaturverwaltung.

Die Stellvertreterklasse ist integriert in die Hierarchie der Literaturklassen und wird von der abstrakten Basisklasse `Reference` abgeleitet.

Grundsätzlich ist dieses Stellvertreterkonzept allgemein auf alle Arten von Objekten in Datenbankkomponenten anwendbar, da alle Objekte über eindeutige Objekt-IDs verfügen. Je nach Anwendung können bei Zugriffen auf Stellvertreterobjekte die enthaltenen Querverweise automatisiert von der Verwaltungsinstanz und damit für die Anwendung transparent verfolgt werden. In diesem Fall ist der Anwendung nicht bekannt, daß Stellvertreterobjekte existieren. Alternativ kann der Anwendung das Stellvertreterobjekt als Ergebnis angeboten werden.

Zugriffsdienst

Der Zugriffsdienst einer Datenbankkomponente stellt Dienste zur Verfügung, die von Clients genutzt werden können. Durch diese zusätzliche Teilkomponente können die darunterliegenden Bereiche (Kontrollinstanz, Verwaltungsinstanz, persi-

stenter Speicher) unabhängig von der eingesetzten Technologie der Kommunikationsschicht entwickelt werden (siehe Abbildung 4.16). Der Zugriffsdienst realisiert die Schnittstelle zwischen Kommunikationsschicht und Datenbankkomponente.

Wird in der Kommunikationsschicht CORBA eingesetzt, werden von Servern exportierte Schnittstellen angebotener Dienste in IDL formuliert. Unabhängig von der konkreten Implementierung einer Datenbankkomponente und ihres Orts im Netzwerk können Clients anhand von IDL-Definitionen Dienste ohne zusätzliches Wissen nutzen.

Angebotene Dienste müssen aus der Sicht der Anwender umfassende Recherchen in den lokalen Datenbeständen ermöglichen. Datenbankkomponenten können daher optional über Rechercheinstanzen verfügen, die komplexe Anfragen optimiert beantworten können. Dies wird notwendig, wenn die von Verwaltungsinstanzen eingesetzten Index-Datenstrukturen nicht umfassend genug sind. Ergebnisse solcher Anfragen sind Listen von Objekt-Identifikatoren. Der Zugriff auf entsprechende Objekte im persistenten Speicher ist nur über Verwaltungsinstanzen möglich. Rechercheinstanzen werden durch Zugriffsdienste ebenfalls nach außen verdeckt: sie sind wie Kontrollinstanzen zwischen Zugriffsdiensten und Verwaltungsinstanzen lokalisiert und damit integraler Bestandteil von Datenbankkomponenten, die nur über exportierte Schnittstellen zugänglich sind. Da Recherchesysteme lediglich lesend auf Datenbestände zugreifen, werden Kontrollinstanzen umgangen.

Neben der Bereitstellung von Diensten der Verwaltungs-, Kontroll- und ggf. Rechercheinstanz, werden vom Zugriffsdienst Informationen gesammelt, die statistische Auswertungen über die Nutzung von Datenbankkomponenten erlauben.

Eingabe neuer Literaturstellen

In die zentrale Literaturverwaltung müssen neue Zitate eingegeben werden. Dies kann grundsätzlich manuell oder automatisiert erfolgen.

Manuelle Eingabe: Bei der manuellen Eingabe müssen Gruppenmitglieder durch geeignete graphische Eingabeformulare unterstützt werden. Diese Formulare müssen alle implementierten unterschiedlichen Literaturtypen anbieten.

Im Rahmen der täglichen Arbeit werden vom wissenschaftlichen Personal interaktive Literaturrecherchen durchgeführt (siehe Abbildung 4.17). Diese erfolgen ausschließlich über das *World-Wide Web*. Relevante Ergebnisse sollen in die lokale Literaturdatenbankkomponente übernommen werden können und dadurch allen Gruppenmitgliedern aller Gruppen zur Verfügung stehen. Es wird gefordert, in HTML dargestellte Zitate unter Zuhilfenahme einer internen Zwischenablage via Maus in die Eingabeformulare der lokalen Literaturdatenbankkomponente eintragen zu können (*copy/paste*). Eventuelle Datenkonvertierungen, aus der HTML

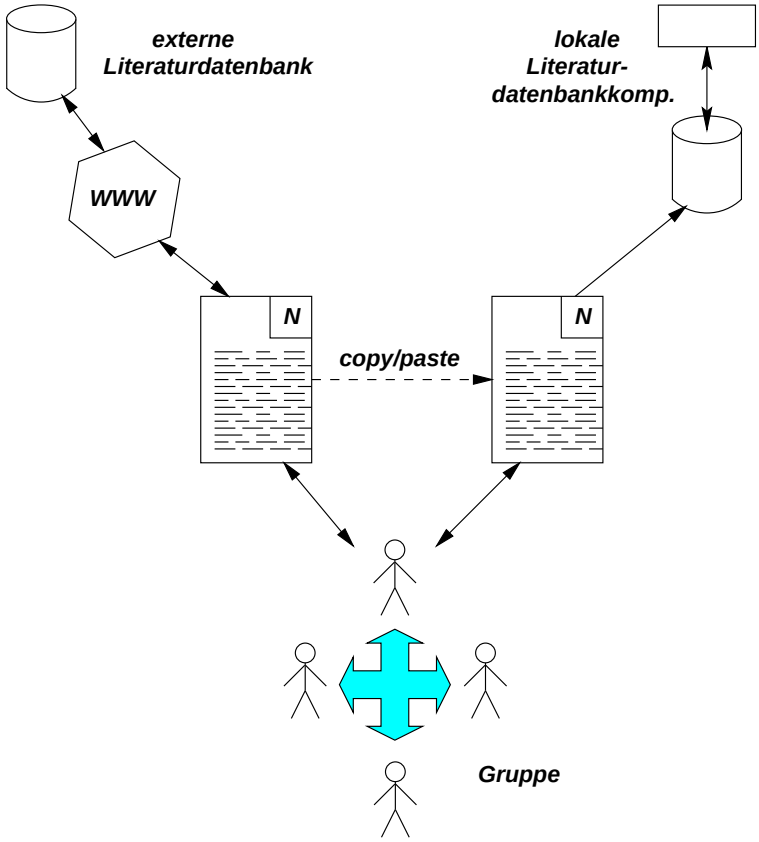

Abbildung 4.17: Manuelle Eingabe neuer Literaturstellen unter Verwendung externer Literaturdienste.

Darstellung in die gruppeninterne Objektpräsentation, werden dabei automatisiert durchgeführt. Solche Formatprobleme werden für Gruppenmitglieder transparent automatisiert gelöst.

Automatisierte Eingabe: Im Rahmen der Integration von Informationen aus externen Quellen werden Literaturzitate automatisiert in die zentrale Literaturverwaltung aufgenommen. Die entsprechenden Daten müssen dazu aus externen Quellen extrahiert und in interne Objektrepräsentationen konvertiert werden. Das Einfügen basiert auf der vorhandenen Infrastruktur (siehe Abbildung 4.18).

Dazu wird ein entsprechender Informationswandler (siehe 4.6) realisiert. Dieser enthält quellenspezifische Import- und Aufbereitungseinheiten. Für die zentrale Literaturverwaltung wurde eine Integrationseinheit entwickelt, die an dieser Stelle wiederverwendet werden kann.

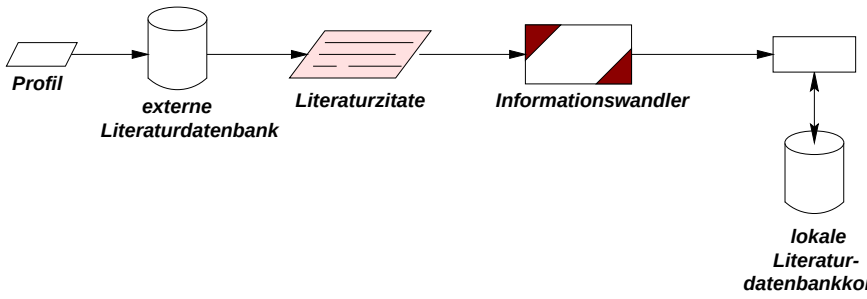

Abbildung 4.18: Automatisierte Eingabe neuer Literaturstellen aus externen Quellen.

Integration von Literaturdiensten: Neben manuellen Literaturrecherchen werden Literaturdienste beauftragt, gemäß aufgestellter Profile wissenschaftliche Quellen nach entsprechender neuer Literatur zu durchsuchen. Resultate solcher Literaturdienste werden Gruppenmitgliedern als Email zugestellt. Diese Ergebnisse sollen automatisiert bearbeitet werden. Die in den Emails enthaltenen Zitate müssen dazu extrahiert und mit der lokalen Literaturverwaltung abgeglichen werden. Die dazu notwendige Anwendung muß die von den Literaturdiensten verwendeten Formate verstehen, um entsprechende Informationen extrahieren und lokale Literaturobjekte generieren zu können. Das Einfügen erfolgt unter Inanspruchnahme des Zugriffsdienstes der Literaturdatenbankkomponente. Redundanzfreiheit wird durch die Kontrollinstanz der Datenbankkomponente gewährleistet und muß von der Anwendung daher nicht beachtet werden. Realisiert wird dies durch literaturdienstspezifische Informationswandler (siehe 4.6).

4.8.2 Spezialisierte Literaturverwaltung

Neben den soeben dargestellten Literaturrecherchen, die von Literaturdiensten durchgeführt werden, sollen Gruppenmitglieder in der Durchführung manueller Literaturrecherchen in öffentlichen Quellen unterstützt werden. Bei der Literaturarbeit muß ein Gruppenmitglied, das an einem bestimmten Thema arbeitet, immer wieder ähnliche Arbeitsschritte durchführen, um über neueste Literatur informiert zu werden. Die erzielten Ergebnisse werden manuell analysiert. Diese alltägliche Arbeit soll von einem Literaturagenten, der als Stellvertreter des Gruppenmitglieds gegenüber einer externen Literaturdatenbank auftritt, automatisiert durchgeführt werden (siehe Abbildung 4.19).

Gruppenmitglieder legen dazu Profile fest, die z.B. aus Listen von Schlagwörtern bestehen können. Anhand dieser Profile führen Literaturagenten regelmäßig Recherchen in den angegebenen öffentlichen Literaturdatenbanken durch.

Erzielte Ergebnisse werden in Datenbanken gesammelt, die spezifisch für Profile angelegt und verwaltet werden. Gruppenmitglieder werden in der Auswertung dieser Ergebnisse unterstützt. So werden z.B. nach Anmeldung eines Gruppenmitgliedes neue Treffer präsentiert. Während der Sichtung der Literaturzitate können Mitglieder entscheiden, ob ein Zitat relevant oder irrelavant ist. Durch geeignete graphische Bedienoberflächen können auf diese Art Zitate durch Mausklick entweder in die zentrale Literaturverwaltung übernommen oder als uninteressant markiert werden. Ein als uninteressant markiertes Zitat darf zu einem späteren Zeitpunkt nicht mehr als neues Zitat vom Agenten angeboten werden. Daher werden entsprechende Informationen in einer spezialisierten Literaturdatenbankkomponente aufgehoben.

Diese speziell auf Profile zugeschnittenen Literaturdatenbanken werden analog der zentralen Literaturverwaltung modelliert. Assoziierte Awareness-Objekte ermöglichen die Entscheidung, ob ein Literaturobjekt neu ist und daher Gruppenmitgliedern angeboten werden muß. Die Entscheidung, daß ein Objekt nicht relevant zum Erreichen des Gruppenziels ist, wird ebenfalls im Awareness-Objekt abgelegt. So kann zu einem späteren Zeitpunkt nachvollzogen werden, welches Mitglied welche Literaturobjekte begutachtet hat. Insbesondere ist es möglich, ursprünglich als uninteressant klassifizierte Zitate zu einem späteren Zeitpunkt als relevante in den gruppenweiten Informationsraum zu übernehmen. Spezialisierte Literaturdatenbanken können projektspezifisch oder auch für individuelle Gruppenmitglieder angelegt werden. Das Sichten erzielter Ergebnisse ist somit auch von einer Gruppe nebenläufig durchführbar. Die in der Kontrollinstanz der Datenbankkomponente realisierte Strategie zur Nebenläufigkeitskontrolle stellt sicher, daß neue Zitate immer nur genau einem Gruppenmitglied zur Begutachtung vorgelegt werden. Die Abarbeitung neuer Treffer kann daher von allen Gruppenmitgliedern arbeitsteilig erledigt werden.

Zur Sichtung der von Agenten erzielten Treffer muß eine geeignete graphische Bedienoberfläche angeboten werden, die die einfache Übernahme in die zentrale Literaturverwaltung durch Mausklick ermöglicht.

Anhand der als interessant eingestuften Zitate sollen Literaturagenten zugrundeliegende Profile überprüfen und ggf. Verbesserungsvorschläge unterbreiten, um genauere und damit effizientere Suchen zu ermöglichen. Diese Verbesserungsvorschläge werden den Gruppenmitgliedern präsentiert, die entsprechende Entscheidungen fällen müssen.

Das Erstellen eines Profils ist oftmals keine einfache Aufgabe. Neue Mitglieder einer Gruppe sind mit Details zugrundeliegender Projekte oftmals noch nicht genug vertraut. Daher sollen Literaturagenten anhand existierender Literatursammlungen eigenständig Profile vorschlagen. Diese Vorschläge können anschließend von Gruppenmitgliedern überarbeitet bzw. angepaßt werden.

4.8.3 Management von Proteinsequenzen

Wie in 3.2.2 dargestellt, werden von einer Untergruppe von MIPS Proteinsequenzen verwaltet. Zusammen mit Partnern aus den USA und Japan wird die Proteinsequenzdatenbank *PIR-International* erstellt, gepflegt, präsentiert und verteilt. Management von Proteinsequenzen bedeutet hier in erster Linie das Einfügen neuer Proteinsequenzen in den gemeinsamen Informationsraum der Proteinsequenzdatenbank sowie die Anreicherung der Rohdaten mit biologischen Zusatzinformationen. Gruppenziel ist, eine möglichst vollständige, redundanzfreie und qualitativ hochwertig annotierte Proteinsequenzdatenbank der wissenschaftlichen Öffentlichkeit anbieten zu können.

Verallgemeinert besteht bei der Kollaboration von *PIR-International* folgende Situation: eine Menge von Gruppen arbeitet in einem gemeinsamen Informationsraum. Da diese Gruppen über verschiedene Kontinente verteilt sind, verfügt jede Gruppe aus Effizienzgründen über lokale Kopien der Objekte des gemeinsamen Informationsraums. Auf diesen Replikaten werden lokal Änderungen durchgeführt. Außerdem werden neue Objekte in den lokalen Informationsraum eingefügt. Im Zuge eines regelmäßigen Datenabgleichs (etwa wöchentlich) werden alle lokalen Objekte aller lokalen Informationsräume miteinander abgeglichen, um Konsistenz der enthaltenen Informationen zu erreichen. Die Gruppen sind untereinander lose gekoppelt: Koordination erfolgt lediglich durch direkte Absprachen und wöchentliche Datenabgleiche.

Für diese Arbeit ist die Unterstützung einzelner Gruppen in ihrer nebenläufigen Arbeit auf dem lokalen Informationsraum interessant. Zur Konsistenzsicherung von Objekten zwischen lokalen Informationsräumen der Partner der Proteinsequenzdatenbank wird auf [Kap95] verwiesen. Im folgenden wird repräsentativ die Annotationsgruppe bei MIPS betrachtet.

Archiv für Proteinsequenzen

Gemäß den Forderungen unter 3.2.2 wird ein Archiv für Proteinsequenzen benötigt. Jede neue Proteinsequenz, die in den Informationsraum der Proteinsequenzdatenbank eingefügt wird, wird dort abgelegt. Im Zuge der Datenpflege können Situationen auftreten, in denen zwei verschiedene Proteinobjekte zu einem zusammengefaßt werden. Durch Einträge im Archiv kann jedoch jederzeit überprüft werden, ob bestimmte Proteinsequenzen in den Informationsraum importiert worden sind.

Dieses Archiv wird analog der Literaturverwaltung als Datenbankkomponente modelliert. Als persistenter Speicher dient eine Datenbank, die von einem DBMS verwaltet wird. Die Verwaltungsinstanz stellt analog der Verwaltungsinstanz der Literaturverwaltung die notwendigen Basisoperationen bereit.

Die Kontrollinstanz gewährleistet Redundanzfreiheit. Für das Archiv bedeutet dies, daß die gleiche Sequenz aus dem gleichen Eintrag einer externen Quelle nicht doppelt eingefügt werden darf.[13] Objekte im Archiv unterliegen keiner Pflege. Allerdings muß die Möglichkeit der Modifikation gegeben werden, um beispielsweise Fehler, die in Quellen existierten und importiert wurden, beheben zu können. Das Archiv muß keine hohe Verfügbarkeit gewährleisten, da Anfragen und Modifikationen eher selten von Gruppenmitgliedern durchgeführt werden. Es wird daher eine pessimistische Strategie der Nebenläufigkeitskontrolle angewendet (siehe 4.2.2).

Annotation

Annotation beinhaltet die Teilaufgaben *Eingabe*, *Pflege* und *Präsentation* biologischer Daten. Zur Realisierung dieser Teilaufgaben vor dem Hintergrund der Gruppenunterstützung müssen beim Entwurf die unter 3.2.2 aufgestellten Forderungen berücksichtigt werden.

Eingabe: Neue Proteinsequenzen werden automatisiert aus öffentlichen Datenbanken in den Gruppen-Informationsraum eingefügt. Für diese Aufgabe wird ein Informationswandler eingesetzt (siehe 4.6). Die zugrundeliegenden externen Datenbanken, die hier als Quelle dienen, zeichnen sich durch komplexe Semantiken aus. Da beim Entwurf der Datenformate viel Wert auf Lesbarkeit gelegt wurde, ist die Entwicklung entsprechender Parser aufwendig. Unabhängig von diesen bei der Implementierung auftretenden Probleme kann ein konkreter Informationswandler inkrementell in das bestehende System übernommen werden. Konsistenzsicherung und Gewährleistung von Redundanzfreiheit wird von der entsprechenden Datenbankkomponente durchgeführt.

Pflege: Objekte im gemeinsamen Informationsraum müssen gepflegt werden. Diese Pflege kann entweder manuell oder automatisiert erfolgen. Zur Konsistenzsicherung wird bei der Proteinsequenzdatenbankkomponente eine pessimistische Strategie zur Nebenläufigkeitskontrolle eingesetzt. Die manuelle Bearbeitung eines Objekts kann mehrere Tage dauern. Gerade vor dem Hintergrund der Telearbeit, die von einigen Gruppenmitgliedern dieser Gruppe wahrgenommen wird, kann nicht davon ausgegangen werden, daß umfangreiche Bearbeitungen von Objekten innerhalb eines Tages durchgeführt werden können. Die Wahrscheinlichkeit, daß sowohl ein Gruppenmitglied als auch ein automatisiert ablaufender Pro-

[13] Das bedeutet insbesondere, daß die Proteinsequenz für sich nicht eindeutig ist. Eine bestimmte Proteinsequenz kann mehrfach im Archiv enthalten sein, vorausgesetzt, die Quellen sind unterschiedlich.

zeß im Rahmen der Pflegearbeiten ein Objekt gleichzeitig bearbeiten wollen, ist daher relativ hoch. Speziell an Wochenenden: um Rechnerressourcen möglichst gut auszunutzen, werden automatisierte Änderungen wochenends durchgeführt. Gruppenmitglieder können jedoch andererseits am Freitag Objekte mit nach Hause genommen haben, um sie in der kommenden Woche wieder in den Gruppen-Informationsraum aktualisiert einzufügen. Da Annotieren, das dieser Pflege zugrundeliegt, eine mitunter aufwendige Tätigkeit ist, die Ergebnisse rechenintensiver Analyseprogramme erfordern kann, ist Sicherstellung der Konsistenz die wichtigste Forderung.

Bei der manuellen Pflege müssen Gruppenmitglieder zu modifizierende Objekte bei der entsprechenden Datenbankkomponente anfordern (siehe Abbildung 4.20). Dabei soll ein Gruppenmitglied jedoch nicht wissen müssen, welche Datenbankkomponente das entsprechende Objekt verwaltet: der Zugriff erfolgt transparent durch Spezifikation einer Objekt-ID.

Ist das angeforderte Objekt verfügbar, wird ein Replikat im persönlichen Informationsraum des Gruppenmitglieds angelegt. Damit liegt es in der Verantwortung dieses Gruppenmitglieds, welche Modifikationen auf dem Objekt durchgeführt werden. Das weitere Kopieren, etwa auf Diskette, um z.B. zu Hause weiterzuarbeiten, ist möglich. Außerdem kann die Verantwortung an ein anderes Gruppenmitglied übertragen werden, um z.B. Spezialwissen eines Kollegen oder einer Kollegin auszunutzen. In diesem Fall migriert das evtl. schon modifizierte Objekt in den entsprechenden persönlichen Informationsraum dieses Mitglieds. Diese Migration wird im Awareness-Objekt vermerkt.[14] Das neu verantwortliche Gruppenmitglied kann nach Abschluß der Modifikationen das Objekt entweder an das ursprüngliche Mitglied zurückgeben oder eigenständig an die Datenbankkomponente zurückreichen.

Allgemein werden nach Abschluß der Bearbeitung Objekte aus persönlichen Informationsräumen in den gemeinsamen Gruppen-Informationsraum zurückgegeben. Die Kontrollinstanz, die die Nebenläufigkeitskontrolle realisiert, stellt sicher, daß das eingereichte Replikat mit dem übereinstimmt, das zu einem früheren Zeitpunkt angefordert wurde. Dadurch wird ausgeschlossen, daß veraltete Objekte, die ein Gruppenmitglied z.B. auf einer Diskette findet, aus Versehen wieder in den Datensatz integriert werden. Diese Gewährleistung basiert auf Zeitstempeln, die im Awareness-Objekt eingetragen werden.

[14]Wird ein Objekt von einem Gruppenmitglied an ein anderes Gruppenmitglied am System vorbei migriert, z.B. via Diskette, kann keine entsprechende Eintragung im Awareness-Objekt durchgeführt werden. Allerdings wird der Fall erkannt, wenn dieses andere Gruppenmitglied ein modifiziertes Objekt bei der Datenbankkomponente einreichen will. Die Absenderinformation und der Vermerk im Awareness-Objekt bzgl. des Gruppenmitglieds, das ein Replikat anforderte, stimmen nicht überein.

Präsentation: Die in den gemeinsamen Informationsräumen enthaltenen anwendungsspezifischen Daten müssen, je nach Zugriffsbestimmungen der jeweiligen Projekte, der wissenschaftlichen Öffentlichkeit präsentiert werden. Die Proteinsequenzdatenbank ist öffentlich, d.h. jedes Objekt ist öffentlich und darf somit von allen Interessierten weltweit eingesehen werden, sobald gruppeninterne Kriterien erfüllt sind.

Um diese Präsentation zu ermöglichen, existieren eine Vielzahl von Möglichkeiten, die angeboten werden sollen. Allgemein kann zwischen der statischen und dynamischen Bereitstellung unterschieden werden.

Bei der *statischen* Variante werden Objekte zu einem bestimmten Zeitpunkt kopiert und anschließend entsprechend aufbereitet. Zugriff wird durch den Dienst `ftp` ermöglicht, durch den Interessierte lokale Kopien der Daten in einem bestimmten Format als Textdatei erhalten können. Alternativ wird der Datenbestand auf einem Massenmedium (CD-ROM, DVD) verschickt. Jedes anwendungsspezifische Objekt muß gemäß der zugrundeliegenden Klassendefinition über Methoden verfügen, die den internen Zustand des Objekts in das entsprechende Format der Bereitstellung exportieren können. Das Erstellen einer entsprechenden Datensammlung kann sehr leicht realisiert werden, indem durch den gesamten Informationsraum navigiert und von jedem besuchten Objekt die entsprechende Exportmethode aufgerufen wird. Auf diese Weise können mehrere unterschiedliche Formate parallel unterstützt werden. Die auf diese Weise erzeugte Kopie der Datensammlung darf inhaltlich nicht mehr modifiziert werden.

Bei der *dynamischen* Alternative der Datenpräsentation kann unterschieden werden, ob diese interaktiv über eine Bedienoberfläche oder aus einem Programm heraus automatisiert erfolgen soll.

Der interaktive Datenzugriff erfolgt über das *World-Wide Web*. Dadurch kann weltweit auf biologische Informationen einer Ressource zugegriffen werden, vorausgesetzt, die entsprechende Browser-Software sowie ein Internetzugang sind vorhanden. Je nach Komplexität werden Informationen in HTML-Seiten oder innerhalb eines Java-Applets, das in der Browser-Software abläuft, präsentiert.

Der dynamische Zugriff aus einer Anwendung heraus, die auf einem beliebigen Rechner im Internet abläuft, soll so einfach durchführbar sein, als wären die Daten lokal vorhanden. Daten werden von der Informationsressource über einen Server angeboten. Als Kommunikationsschicht empfiehlt sich an dieser Stelle erneut der OMG-Standard CORBA. Nach Bereitstellen der IDL-Definitionen durch den Dienstanbieter können dem Server nicht bekannte Clients die von ihm exportierten Schnittstellen nutzen, indem gemäß der Schnittstellen Clients entwickelt werden. Auch aus einem Java-Applet heraus, das zwar der Datenbankbetreiber entwickelt hat, das aber beim entfernten Client abläuft, kann der Datenzugriff basierend auf CORBA transparent realisiert werden.

Klassifikation von Proteinsequenzen

Durch die Klassifikation von Proteinsequenzen werden Zusammenhänge zwischen Proteinobjekten im gemeinsamen Informationsraum hergestellt. Aufgrund der hierarchischen Einteilung in Familien und Superfamilien (vgl. 3.2.2) erfolgt außerdem eine Strukturierung der Objekte. Die Klassifikationen erfolgen anhand der Ähnlichkeit von Proteinsequenzen. Dazu werden entsprechende Algorithmen angewendet (siehe [BGM+99]).

Die Klassifikationsinformationen werden durch eine eigenständige Datenbankkomponente modelliert. Auf diesem Informationsraum werden unter Einsatz entsprechender Analysealgorithmen inkrementell Klassifikationsinformationen verwaltet. Erzielte Ergebnisse werden in die Proteinobjekte der Proteinsequenzdatenbankkomponente automatisiert durch den oben beschriebenen Mechanismus integriert. Da die Durchführung der Klassifikation fast ausschließlich automatisiert und von einem einzelnen Gruppenmitglied duchgeführt wird, wird dieser Aspekt hier nicht näher betrachtet.

4.8.4 Systematische Genomsequenzierungsprojekte

Von den in 3.2 dargestellten Gruppen werden die Gruppen, die an systematischen Genomsequenzierungsprojekten beteiligt sind, im prototypischen Groupwaresystem nur partiell unterstützt. Bei der Rechnerunterstützung wird nur der Teil betrachtet, der die Verbindung zur oben beschriebenen Proteinsequenzdatenbank darstellt.[15] Haben Daten innerhalb dieses Projekts einen Zustand erreicht, der die Publikation der Informationen erlaubt, werden Sequenzdaten und ihre Zusatzinformationen sowohl bei der zuständigen Nukleinsäuredatenbank, als auch bei der hier beschriebenen Proteinsequenzdatenbank eingereicht. Für diese Aufgabe sollen die in diesem Projekt arbeitenden Gruppenmitglieder durch das CSCW-System unterstützt werden.

Die im Sequenzierungsprojekt vom Informatikkoordinator gesammelten und aufbereiteten Daten werden als Objekte in einer organismusspezifischen Datenbankkomponente verwaltet (siehe Abbildung 4.21). Zu einem geeigneten Zeitpunkt wird der Export der Daten durch ein Gruppenmitglied angestoßen. Für die Nukleinsäuredatenbank bedeutet dies, daß die durch die Objekte repräsentierten biologischen Informationen entsprechend den Anforderungen des externen Datenbankbetreibers als Textdateien in definiertem Format darzustellen sind. Dies wird durch einen Informationswandler ermöglicht.

[15] Wie eine allgemeine Unterstützung für diese Art von Projekten in der Molekularbiologie durch ein Managementsystem aussehen kann, wird derzeit in einer Dissertation bei MIPS erarbeitet.

Für die Integration der Daten in den Informationsraum der Proteinsequenzdatenbank wird analog der Darstellung in 4.6 ein entsprechender Informationswandler erstellt. Die Importeinheit des Informationswandlers kann dabei direkt auf Methoden der Objekte über die beschriebene Infrastruktur zugreifen. Die Konvertierung in die Proteinobjekte kann von der Aufbereitungseinheit in dieser Situation vollständig automatisiert erfolgen, da zwischen beiden Gruppen Absprachen in Bezug auf die Dateninhalte (Semantik) durchgeführt wurden. Die Integrationseinheit schließlich fügt über die Kommunikationsschicht unter Anwendung des Zugriffsdienstes der Proteinsequenzdatenbankkomponente neue Objekte in den gemeinsamen Gruppen-Informationsraum ein.

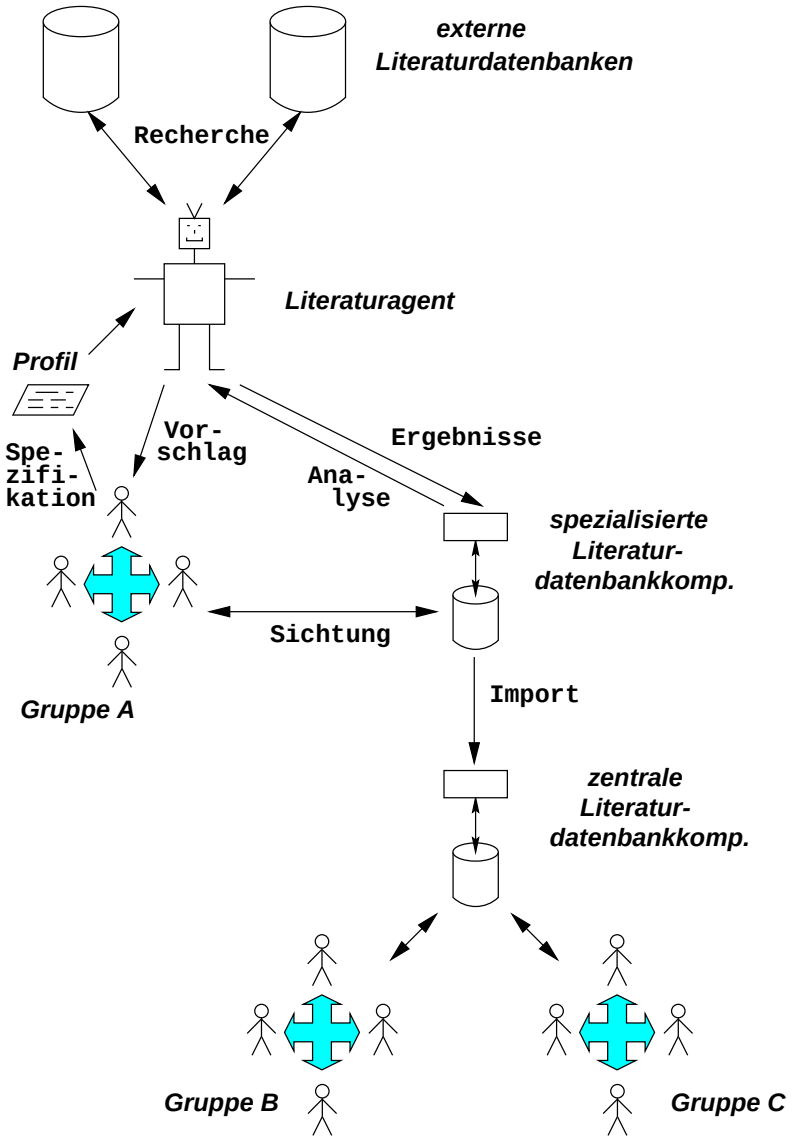

Abbildung 4.19: Spezialisierte Literatursammlungen unter Einsatz von Literaturagenten.

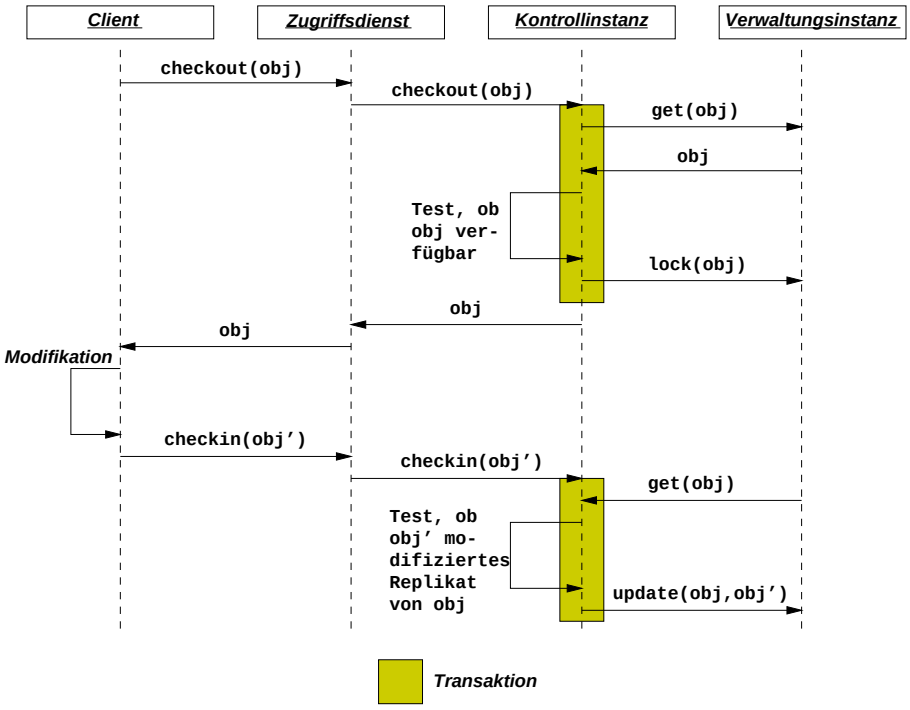

Abbildung 4.20: Pessimistische Variante zur Nebenläufigkeitskontrolle.

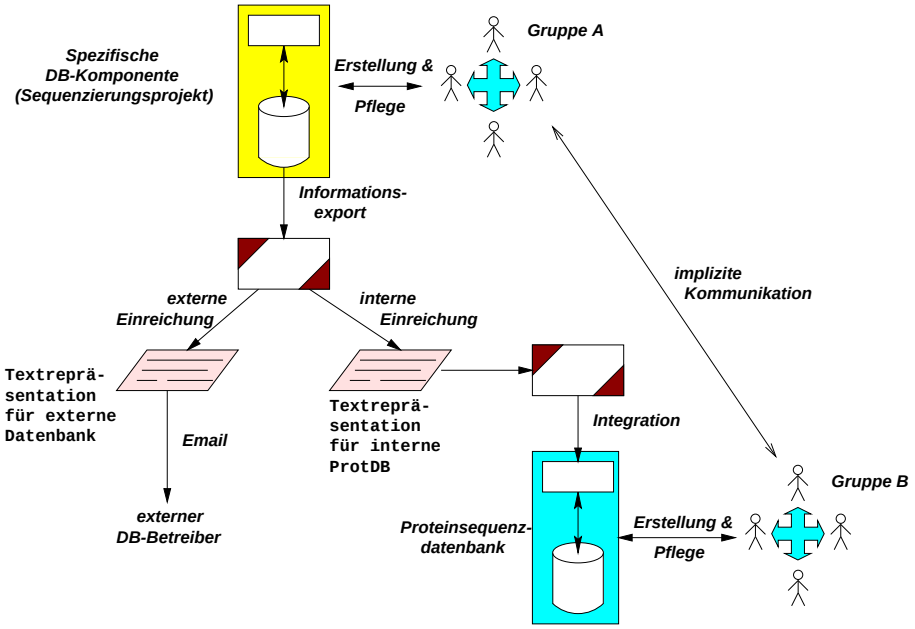

Abbildung 4.21: Automatisierter Informationsexport aus projektspezifischen Informationsräumen in externe bzw. lokale Informationsräume.

Kapitel 5

Realisierung des entworfenen Groupwaresystems

In diesem Kapitel wird auf Implementierungsaspekte des prototypisch entwickelten Groupwaresystems eingegangen. Gemäß den Entwürfen des vorangegangenen Kapitels wird zunächst die allgemeine Realisierung von Datenbankkomponenten, der Kommunikationsschicht sowie von Informationswandlern dargestellt. Spezielle Aspekte ausgewähler Applikationen schließen sich an.

5.1 Datenbankkomponente

Eine Datenbankkomponente ist eine Einheit im Groupwaresystem mit folgenden Aufgaben:

- Verwaltung von Objekten eines Anwendungsobjektmodells.
- Realisierung einer Strategie zur Nebenläufigkeitskontrolle.
- Evtl. Zusicherung von Redundanzfreiheit.
- Bereitstellung von Zugriffsmöglichkeiten auf persistente Objekte gemäß der eingesetzten Kommunikationstechnologie.

Zur Verwaltung aller Objekte des Anwendungsmodells können mehrere Datenbankkomponenten eingesetzt werden. Das einer konkreten Komponente zugrundeliegende Objektmodell deckt daher u.U. nur einen gewissen Teilbereich des Anwendungsgebiets ab.

Eine Datenbankkomponente besteht (gemäß Definition, siehe 4.3.2) aus den verschiedenen Untereinheiten *Persistenter Speicher, Verwaltungsinstanz, Kontrollinstanz* und *Zugriffsdienst*, die in ihrer Gesamtheit die Funktionalität realisieren.

5.1.1 Persistenter Speicher

Einsatz eines DBMS

Im persistenten Speicher werden Anwendungsobjekte abgelegt. D.h. diese Objekte existieren länger als der Prozeß, der sie erzeugt hat (Definition *Persistenz*). Die Organisation auf systemnaher Ebene wird durch kommerziell verfügbare Datenbankmanagementsysteme (DBMS) übernommen. Da auf Modellierungsebene Objekte entworfen werden, muß das eingesetzte DBMS in der Lage sein, diese Objekte in einer Datenbank abzubilden.

Im wesentlichen gibt es derzeit zwei Typen von DBMS: Bei *relationalen* Datenbanken erfolgt die Ablage der Anwendungsdaten in Tabellen gemäß einem Entity-Relationship-Modell, bei *objektorientierten* Datenbanken werden Objekte gespeichert. Liegt einer Anwendung ein Objektmodell zugrunde und soll ein relationales DMBS (RDBMS) zum Einsatz kommen, muß eine Änderung der Informationsrepräsentation von Objekten in Tabellen erfolgen. Dabei müssen Techniken der objektorientierten Welt, wie z.B. Vererbung und Polymorphismus, entsprechend abgebildet werden. Es ist zu beachten, daß Objektidentität besteht: zwei Objekte, die sich in ihren Zuständen nicht unterscheiden, unterscheiden sich in ihrer Identität. In relationalen Datenbanken unterscheiden sich zwei Einträge nur durch ihre Werte. Einfache Änderungen am Objektmodell, etwa das Ableiten einer neuen Klasse von einer bestehenden, führen zu keinen weitergehenden Änderungen an der objektorientierten Datenbank. Liegt eine relationale Datenbank zugrunde, können umfangreiche Reorganisationen der Tabellen die Folge sein.

Das OODBMS *ObjectStore*

Aufgrund der Dynamik und der komplexen Objekte dieses Anwendungsgebiets sowie der vielfältigen Abhängigkeiten zwischen den Objekten, wird ein OODBMS bevorzugt. Es wird *ObjectStore* von Object Design eingesetzt, das als C++ Klassenbibliothek vorliegt.[1] Objektorientierte Datenbanken verfügen über kein explizites Schema. Vielmehr wurde durch das Objektmodell bereits festgelegt, welche Objekte in der Datenbank verwaltet werden. Änderungen bzw. Erweiterungen erfolgen zunächst am Objektmodell. Werden existierende Klassen modifiziert, müssen persistente Objekte, die Instanzen dieser Klassen darstellen, entsprechend angepaßt werden. Dazu werden von objekt-orientierten Datenbankmanagementsystemen i.d.R. Schema-Evolutionswerkzeuge bereitgestellt. Diese erlauben entsprechende Anpassungen bereits in Datenbanken existierender Objekte. Neue Klassen können inkrementell hinzugefügt werden. Bereits existieren-

[1] Mittlerweile wird auch Java unterstützt. Sowohl C++- als auch Java-Applikationen können auf die gleiche Datenbank zugreifen.

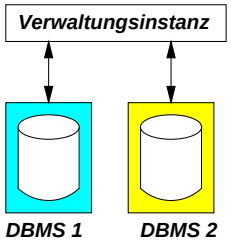

Abbildung 5.1: Die physikalische Ablage der Daten kann unter Einsatz unterschiedlicher Datenbankmanagementsysteme erfolgen. Diese DBMS werden durch die Verwaltungsinstanz gekapselt. Eine Reorganisation der Datenbanken bzw. Datenbankmanagementsysteme führt lediglich zu Veränderungen in der Verwaltungsinstanz. Hierarchisch höher angesiedelte Einheiten einer Datenbankkomponente sind davon nicht betroffen.

de Objekte sind davon nicht betroffen.

ObjectStore ist als verteiltes System realisiert. Die Datenbank wird von einem Server verwaltet. Anwendungen, die auf Datenbanken zugreifen, sind Datenbankclients. Zur Performanzsteigerung verfügt jede Maschine, auf der ein Datenbankclient läuft, über einen Zwischenspeicher (*cache*), der einen effizienten Zugriff auf bereits geladene Seiten ermöglicht. Eine Verwaltungseinheit (*cache manager*) gewährleistet Konsistenz zwischen einem lokalen Zwischenspeicher und der Datenbank beim Server.

5.1.2 Verwaltungsinstanz

Die Verwaltungsinstanz verwaltet den persistenten Speicher, d.h. regelt die interne Organisation der Anwendungsobjekte in Datenbanken des DBMS. Das anwendungsspezifische Objektmodell wird unabhängig vom eingesetzten DBMS entwickelt. Der datenbankspezifische Teil, insbesondere DBMS abhängiger Quellcode, wird in der Verwaltungsinstanz gebündelt. Es erfolgt dadurch eine Kapselung. Soll das eingesetzte DBMS ausgestauscht werden, ist lediglich dieser Teil einer Datenbankkomponente entsprechend zu verändern.

Darüberhinaus kann die physikalische Ablage der Daten von mehr als einem DBMS realisiert sein (siehe Abbildung 5.1). Die Verwaltungsinstanz bedient sich dabei der Funktionalitäten dieser DBMS, die sie integriert.

Neben Basisoperationen, wie `insert`, `remove`, `get` und `update`, werden je nach Anwendung weitere Operationen angeboten (z.B. `checkout`, `checkin`, `merge`, etc.) Der Zugriff auf persistente Objekte erfolgt innerhalb von Transaktionen, die die ACID-Forderungen erfüllen (vgl. 4.2.1).

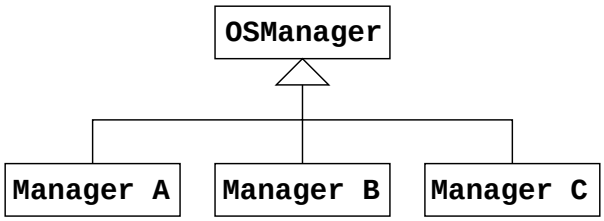

Abbildung 5.2: Datenbankkomponenten verfügen über spezialisierte Managerklassen, die die Verwaltung des persistenten Speichers durch Kommunikation mit den jeweils eingesetzten Datenbankmanagementsystemen durchführen. Allgemeine Funktionalitäten im Zusammenhang mit dem objektorientierten Datenbankmanagementsystem *ObjectStore* sind in die allgemeine Basisklasse `OSManager` ausgelagert, von der alle konkreten Managerklassen abgeleitet werden. Der OODBMS spezifische Quellcode ist in den Managerklassen gekapselt. Außerhalb dieser Klassen muß kein Wissen über das eingesetzte DBMS vorhanden sein.

Zur Realisierung der Verwaltungsinstanz wird ebenfalls ein Objektmodell erzeugt. Bei der Analyse der erforderlichen Funktionalitäten können allgemeine Teile in einer Basisklasse gekapselt werden. Allen konkreten Verwaltungsinstanzen liegen Klassen zugrunde, die von dieser Basisklasse abgeleitet werden.

Die Basisklasse `OSManager`,[2] die *ObjectStore*-spezifische Basisklasse, realisiert Methoden zum Anlegen und Öffnen von Datenbanken sowie zum Erzeugen von Wurzelobjekten[3] und ihrem Zugriff. Abgeleitete Klassen werden allgemein als *Datenbankmanagerklassen* bezeichnet (siehe Abbildung 5.2). Diese Managerklassen sind abhängig vom Objektmodell der entsprechenden Anwendung. Die Methode `insert` z.B. erwartet Objekte des Objektmodells der entsprechenden Anwendung als Parameter.

Neben dem Bereitstellen dieser Basisoperationen verwaltet die Verwaltungsinstanz optional Indexdatenstrukturen. Diese ermöglichen einen optimierten Zugriff auf Objekte. Eine Navigation durch die gesamte Datenbank aller Objekte wird dadurch vermieden, um spezifizierte Anfragekriterien zu überprüfen. Die eingesetzten Datenstrukturen sind dabei abhängig vom anwendungsspezifischen Objektmodell sowie den erwarteten Anfragen. Gegenüber RDBMS, denen mit der Relationenalgebra ein mathematisches Modell zugrundeliegt, ist bei OODBMS ein Mehraufwand im Hinblick auf die optimierte Ausführung gestellter Anfragen zu

[2] `OS` steht für `ObjectStore`.

[3] Wurzelobjekte realisieren Einstiegspunkte in den persistenten Speicher. Anwendungsobjekte werden unter diesen Wurzeln eingefügt. Den Wurzelobjekten werden von den Anwendungsapplikationen Namen zugeteilt, über die ein Zugriff von allen Prozessen aus ermöglicht wird.

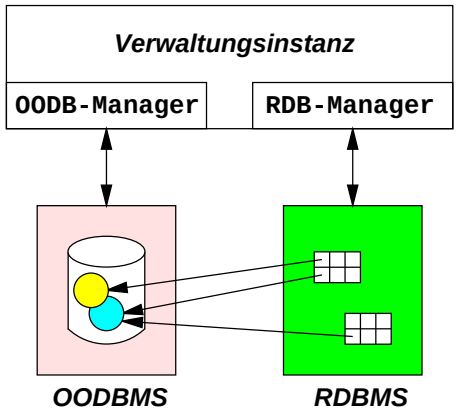

Abbildung 5.3: Indexdatenstrukturen zu einer objektorientierten Datenbank können unter Einsatz eines RDBMS verwaltet werden. Ergebnisse einer Anfrage an eine Indexdatenbank sind Verweise auf Objekte in der objektorientierten Datenbank.

leisten. Grundsätzlich kann bei einer Anfrage die gesamte Semantik der Objekte ausgenutzt werden. Allerdings ist aus Performanzgründen dies nicht praktikabel, falls die Anzahl der Objekte eine kritische Größe überschreitet (abhängig von der Komplexität der Objekte sowie der Leistung der eingesetzten Rechner).

Indexdatenstrukturen können ebenfalls als Objekte in der gleichen Datenbank abgelegt werden, die auch die Anwendungsobjekte enthält. Alternativ ist der Einsatz eines RDBMS denkbar. Zu einer objektorientierten Datenbank existieren eine Menge relationaler Datenbanktabellen, die von einem RDBMS verwaltet werden (siehe Abbildung 5.3). Je nach Anwendung werden entsprechende Informationen der Anwendungsobjekte redundant in Tabellen relationaler Datenbanken abgelegt. Zur Beantwortung komplexer Anfragen steht dann die gesamte Funktionalität der relationen Datenbanken (inkl. Anfragesprachen wie etwa SQL) zur Verfügung.

Diese Strategie bietet sich an, wenn Anwendungsobjekte eine hohe Komplexität aufweisen, die objektorientierte Datenbank viele Objekte enthält und das Anfragespektrum groß ist.

Um diese Strategie im Modell abbilden zu können, wird die Verwaltungsinstanz entsprechend erweitert. Neben objektorientierten Datenbankmanagern existieren zusätzlich relationale Datenbankmanager. Komplexe Anfragen werden zunächst an den relationalen Datenbankmanager übergeben. Dieser liefert als Ergebnis eine Liste von Objekt-IDs. Über diese Liste können unter Einsatz des objektorientierten Datenbankmanagers die Anwendungsobjekte in der objektorientierten Datenbank adressiert werden.

Bei modifizierenden Operationen auf dem Datensatz muß Konsistenz zwi-

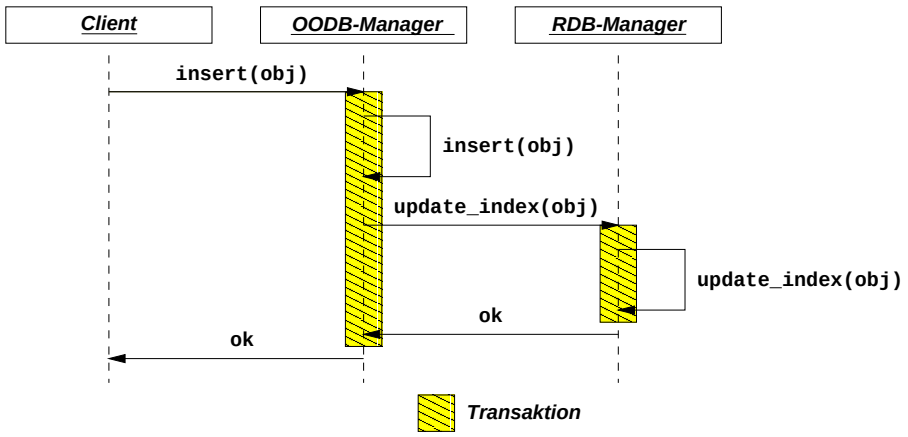

Abbildung 5.4: Werden mehrere Datenbanken zur Verwaltung von Informationen eingesetzt, muß Konsistenz zwischen diesen redundanten Daten sichergestellt sein. Bei modifizierenden Operationen, wie z.B. der Einfüge-Operation, werden daher Transaktionen der jeweiligen DBMS geschachtelt. Im Fehlerfall können Transaktionen von innen nach außen abgebrochen bzw. bereits durchgeführte Transaktionen rückgängig gemacht werden.

schen allen beteiligten Datenbanken sichergestellt werden: zwischen Objekten in objektorientierten Datenbanken sowie ihren assoziierten relationalen Indexdatenbanken. Aktualisierungen relationaler Datenbanken erfolgen innerhalb von Transaktionen. Diese Transaktionen sind in der Transaktion der objektorientierten Datenbank enthalten (siehe Abbildung 5.4). Im Fehlerfall können diese geschachtelten Transaktionen abgebrochen werden, ohne daß Inkonsistenzen entstanden sind.

Liegt eine erweiterte Verwaltungsinstanz vor, kann in die Datenbankkomponente optional eine Rechercheinstanz eingefügt werden.

5.1.3 Kontrollinstanz

Die Kontrollinstanz einer Datenbankkomponente realisiert eine Strategie zur Nebenläufigkeitskontrolle und stellt Redundanzfreiheit sicher, falls diese Anforderung besteht. Da zur Umsetzung der erforderlichen Funktionalitäten Zugriffe auf die Datenbank durchgeführt werden müssen, wird auf Methoden des entsprechenden Manager-Objekts zurückgegriffen, das die Verwaltungseinheit realisiert.

Einer Kontrollinstanz werden von der Verwaltungsinstanz spezielle Varianten von Basisoperationen bereitgestellt. Zur Realisierung von Nebenläufigkeitsstrategien sowie der Gewährleistung von Redundanzfreiheit sind ggf. eine Rei-

he von Anfragen an die Datenbank innerhalb einer Operation (wie z.B. `insert`) notwendig. Wie oben erwähnt, ist der Zugriff auf persistenten Speicher nur innerhalb von Transaktionen möglich. *ObjectStore* erlaubt zwar die Schachtelung von Transaktionen, allerdings müssen alle Transaktionen vom gleichen Typ sein (d.h. entweder *lesend* oder *schreibend*). Für die Realisierung etwa der `insert`-Operation wird jedoch zur Sicherstellung der Redundanzfreiheit auf `get`-Operationen zurückgegriffen: bevor ein neues Objekt eingefügt wird, wird zunächst überprüft, ob es bereits in der Datenbank enthalten ist. Dazu werden `get`-Operationen aufgerufen. `insert`-Operationen werden innerhalb schreibender Transaktionen ausgeführt, `get`-Operationen hingegen innerhalb lesender. In der dargestellten Situation würde es zu einem Laufzeitfehler kommen, da innerhalb der schreibenden `insert`-Transaktion die lesende `get`-Transaktion ausgeführt werden müßte.

Die einfache serielle Abfolge von `get`- und `insert`-Methoden innerhalb jeweils abgeschlossener Transaktionen sichert keine Redundanzfreiheit zu. Es könnte der Fall eintreten, daß zwischen diesen beiden Transaktionen ein weiterer Prozeß eine `insert`-Methode ausgeführt und damit den Zustand der Datenbank verändert hat. Nach Abschluß dieser beiden `insert`-Transaktionen können Redundanz bzw. Inkonsistenzen nicht ausgeschlossen werden. Daher bietet die Verwaltungsinstanz Varianten der Basisoperationen der Kontrollinstanz an, die keine Transaktionen enthalten. Die Verantwortung der Transaktionsgrenzen wird auf die Kontrollinstanz abgewälzt, die alle Datenbankoperationen innerhalb nur einer Transaktion ausführt. Eine fehlerhafte Schachtelung ist dadurch ausgeschlossen.

Nebenläufigkeitskontrolle

Wie in 4.2.2 dargestellt, kommen zwei Strategien der Nebenläufigkeitskontrolle zum Einsatz.

Pessimistische Variante Bei der pessimistischen Variante wird das Entstehen von Inkonsistenzen vermieden. Gemäß dem hybriden Ansatz zur Datenhaltung wird schreibender Zugriff auf Informationsobjekte durch eine zentrale Instanz koordiniert. Diese Aufgabe übernimmt die Kontrollinstanz einer Datenbankkomponente.

Schreibender Zugriff wird serialisiert. Bevor ein Prozeß ein persistentes Objekt modifizieren kann, muß eine transiente Kopie (Replikat) des entsprechenden Objekts von der Datenbankkomponente angefordert werden (*checkout*). Ist das Objekt verfügbar, d.h. es wurde von keinem weiteren Prozeß zu einem früheren Zeitpunkt angefordert und nicht wieder der Kontrollinstanz übergeben, wird ein transientes Replikat erzeugt. Das persistente Original wird als gesperrt markiert.

Im Awareness-Objekt des Anwendungsobjekts werden entsprechende Informationen eingetragen: Zeitstempel der *checkout* Anforderung, Name des Gruppenmitglieds bzw. Prozeßname bei automatisierten Modifikationen. Die genannten Operationen (Test, ob Objekt verfügbar und setzen der Sperre inkl. Aktualisierung des Awareness-Objekts) werden innerhalb einer schreibenden Transaktion ausgeführt. Dadurch wird sichergestellt, daß kein nebenläufiger Prozeß zwischen diesen beiden Operationen zusätzlich eine Kopie des persistenten Objekts erhält.

Das Replikat wird über die Kommunikationsschicht in den persönlichen Informationsraum des Clients transportiert. Dort kann das Objekt bearbeitet werden. Nach Abschluß der Modifikationen wird das Objekt wieder an die Datenbankkomponente übergeben (*checkin*). Die Kontrollinstanz überprüft nun anhand der Zeitstempel in den Awareness-Objekten, ob das eingereichte Objekt ein Replikat des persistenten Objekts ist. Ist dies der Fall, wird ggf. datenbankweit auf Redundanzfreiheit getestet (s.u.). Dadurch wird ausgeschlossen, daß aufgrund von Modifikationen redundante Informationen in den gemeinsamen Informationsraum eingebracht wurden.

Waren alle Tests erfolgreich, wird das persistente Objekt durch das modifizierte Replikat ersetzt, das Awareness-Objekt aktualisiert (z.B. Zeitstempel der letzten Veränderung) und die Sperre aufgehoben. Das Objekt steht für Modifikationen wieder zur Verfügung.

Ist ein Objekt nicht verfügbar, werden die Awareness-Informationen aus diesem Objekt dem Anforderer präsentiert. Dieser kann entsprechend reagieren, indem er z.B. das Gruppenmitglied, das das Objekt gerade bearbeitet, kontaktiert (direkt, per Email oder per Telefon).

Die pessimistische Strategie zur Nebenläufigkeitskontrolle kann unabhängig vom zugrundeliegenden anwendungsspezifischen Objektmodell realisiert werden. Die für die Funktionalität der Nebenläufigkeitskontrolle notwendige Information wird durch das Awareness-Objekt, das jedes persistente Objekt sowie jedes transiente Replikat enthält, repräsentiert.

Optimistische Variante Bei der optimistischen Variante wird davon ausgegangen, daß keine Inkonsistenzen entstehen. Daher kann zu jedem Zeitpunkt ein Objekt zur Modifikation angefordert werden (*checkout*). Nach Abschluß der Modifikationen wird das Replikat der Datenbankkomponente wieder übergeben (*checkin*).

Bevor Änderungen persistent übernommen werden können, muß eine Konfliktanalyse durchgeführt werden. Diese erfolgt über Zeitstempel im Awareness-Objekt beteiligter Objekte bzw. Replikate: Bei der Ausführung des *checkout* Befehls wird in das Awareness-Objekt des Replikats der *last-modified* Zeitstempel des persistenten Objekts eingetragen. Beim *checkin* wird dieser Zeitstempel des

Replikats mit dem *last-modified* Zeitstempel des persistenten Objekts verglichen. Stimmen diese überein, kam es in der Zwischenzeit zu keinen Modifikationen. In diesem Fall können die Änderungen in die Datenbank übernommen werden.

Sind sie unterschiedlich, wurde das persistente Objekt in der Zwischenzeit modifiziert. In diesem Fall muß eine Konfliktanalyse durchgeführt werden. Verfügt die Datenbank über eine *history*, d.h. zu jedem Objekt existiert eine Liste älterer Versionen, kann anhand dieser Objektversionen, dem aktuellen persistenten Objekt sowie dem transienten Replikat, je nach Komplexität der anwendungsspezifischen Semantik der Objekte, auf syntaktischer Ebene entschieden werden, ob Änderungen im Replikat einfach übernommen werden können[4] oder ob Gruppenmitglieder über die Situation informiert werden und manuell entscheiden müssen.

Kann ein Konflikt nicht automatisiert behoben werden, müssen Gruppenmitglieder über diese Situation benachrichtigt werden. Trat ein Konflikt im Rahmen einer interaktiven Aktion auf, kann diese Notifikation direkt erfolgen. Idealerweise wird dem Gruppenmitglied die Situation graphisch aufbereitet präsentiert, indem Unterschiede in den Objektversionen hervorgehoben werden. Diese Präsentation sollte entsprechende Handlungen zur Konfliktbehebung ermöglichen.

Erfolgte der Konfliktfall im Rahmen automatisiert ablaufender Prozesse, müssen Zustände in entsprechenden Logdateien vermerkt werden. Nach Abschluß des Prozesses muß ein Gruppenmitglied, etwa das, das den Prozeß angestoßen hat, über solche Situationen informiert werden. Das Gruppenmitglied muß analog der interaktiven Anwendung unterstützt werden, die Liste der Konfliktfälle bearbeiten zu können.

Gewährleistung der Redundanzfreiheit

Die Forderung nach Redundanzfreiheit ist abhängig von der jeweiligen Anwendung. Redundanzfreiheit bedeutet, daß ein Objekt als Repräsentant einer anwendungsspezifischen Informationseinheit nur einmal im Datensatz enthalten sein darf. So darf etwa bei der zentralen Literaturverwaltung ein Zitat eines wissenschaftlichen Artikels nur einmal eingefügt werden, auch wenn z.B. von zwei unterschiedlichen Quellen unterschiedliche Abkürzungen für ein bestimmtes wissenschaftliches Journal verwendet werden.

Um Redundanzfreiheit sicherstellen zu können, müssen vor dem Einfügen eines neuen Objekts bzw. vor der Übernahme von Modifikationen aus einem Replikat entsprechende Analysen in dem gemeinsamen Informationsraum durchgeführt werden. Je nach Anwendung existieren unterschiedliche Kriterien. D.h. die Gewährleistung der Redundanzfreiheit ist nur auf semantischer Ebene realisierbar, da Inhalte der Anwendungsobjekte betroffen sind. Daher muß dieser Aspekt

[4]Dies ist z.B. der Fall, wenn zwei Gruppenmitglieder unabhängig voneinander verschiedene Bereiche eines Objekts bearbeitet haben.

bei jeder konkreten Datenbankkomponente anhand des zugrundeliegenden Objektmodells geeignet entworfen und implementiert werden. Die dazu notwendige Infrastruktur kann generisch bereitgestellt werden.

5.1.4 Zugriffsdienst

Der Zugriffsdienst exportiert Schnittstellen, die von Clients importiert werden können. Um die interne Organisation bestehend aus Verwaltungs-, Kontroll- und ggf. Rechercheinstanz nach außen zu verdecken, wird der Zugriffsdienst als eigenes Objekt modelliert, das mit den darunterliegenden Managerobjekten kommuniziert. Der Zugriffsdienst ist abhängig von der eingesetzten Kommunikationstechnologie. Erfolgt der Einsatz von CORBA, wird der Zugriffsdienst als CORBA-Server realisiert. Werden andere Protokolle verwendet, müssen entsprechende Zugriffsdienste angeboten werden.

Änderungen an der internen Organisation der Objekte in Datenbanken sollten keine Änderungen an den exportierten Schnittstellen zur Folge haben, außer die Funktionalität wird verändert bzw. erweitert. Dadurch können den Zugriffsdiensten i.d.R. nicht bekannte Clients trotz Datenbank-interner Reorganisationen unverändert ablaufen. Werden neue Schnittstellen angeboten, müssen nur die Clients entsprechend angepaßt werden, die diese neue Funktionalität nutzen wollen. Auf Abwärtskompatibilität muß bei jeder Modifikation exportierter Schnittstellen geachtet werden, da sich ansonsten eine Reihe von Anpassungen ergäben. Diese Situation wird im Kontext von CORBA dadurch erschwert, daß die entwickelten Clients von den jeweiligen Entwicklern angepaßt werden müßten. Diese sind jedoch (i.d.R.) nicht bekannt, so daß sie auch nicht über die Veränderungen informiert werden können.

Aus der Sicht eines Clients sind von einer Datenbankkomponente lediglich die exportierten Schnittstellen sichtbar (im Fall von CORBA als IDL-Definitionen). Die Realisierung der Ortstransparenz wird der Verantwortung der Kommunikationsschicht übergeben.

5.2 Kommunikationsschicht

Aufgabe dieser Schicht ist die Kommunikationsunterstützung in einer heterogenen Umgebung mit dem Ziel, Clients ortstransparenten Zugriff auf Serverfunktionalitäten über Plattform- und Programmiersprachengrenzen hinweg zu ermöglichen.

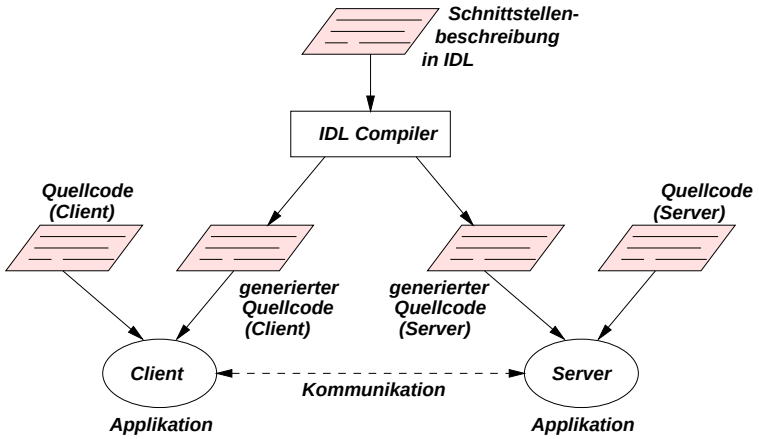

Abbildung 5.5: Schnittstellen werden in CORBA in der Schnittstellenbeschreibungssprache *IDL* verfaßt. Spezielle Übersetzer erzeugen aus diesen Beschreibungen programmiersprachenspezifischen Quellcode, der zu den Applikationen hinzugefügt wird. Dieser generierte Quellcode realisiert u.a. die Kommunikation zwischen Client und Server im CORBA Umfeld.

5.2.1 CORBA

CORBA wird in diesem Groupwaresystem als Kommunikationstechnologie eingesetzt. Schnittstellen von Datenbankkomponenten werden in der Schnittstellenbeschreibungssprache *IDL* spezifiziert. Übersetzer erzeugen daraus programmiersprachenspezifischen Quellcode, der die Kommunikation gemäß dem CORBA-Standard realisiert (siehe Abbildung 5.5). Die Implementierung der Zugriffsdienste erfolgte bei den entwickelten Datenbankkomponenten in der Programmiersprache *C++*.

Neben der plattform- und programmiersprachenunabhängigen Beschreibung exportierter Schnittstellen erlaubt CORBA zur Laufzeit transparente entfernte Objektzugriffe. Die Vermittlung zwischen Client und Server erfolgt durch einen *Object Request Broker (ORB)* (siehe Abbildung 5.6). CORBA-Server melden Schnittstellen, die für Clients bereitstehen, bei einem *ORB* an. Dieser verfügt über interne Verzeichnisse (*interface repository*), in die Schnittstellen registrierter Server eingetragen werden. Zur Laufzeit werden Anfragen von Clients zunächst an den Vermittler *ORB* geschickt. Dieser ermittelt anhand der angeforderten Schnittstelle und seinen Einträgen im *interface repository* einen geeigneten Server. Die Clientanfrage wird an diesen Server weitervermittelt, das erzielte Ergebnis an den Client zurückgegeben. D.h. zwischen Client und Server wird eine indirekte Kommunikation realisiert.

Die im CORBA-Standard enthaltenen Protokollspezifikationen ermöglichen

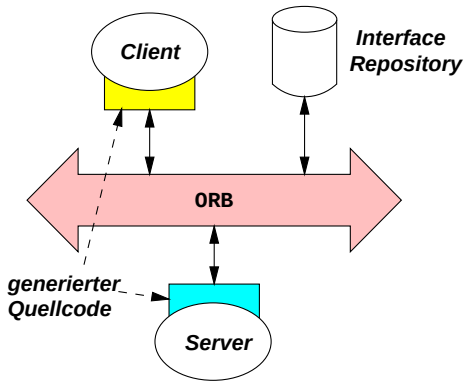

Abbildung 5.6: Die Vermittlung zwischen Client und Server erfolgt durch *Object Request Broker (ORB)*. Server melden ihre Dienste beim ORB an, der diese Informationen Clients transparent zur Verfügung stellt.

die Kommunikation zwischen CORBA-Komponenten, die von verschiedenen CORBA-Produkten (und -Herstellern) erzeugt wurden. Abgesehen von der Spezifikation der Semantik, die in Prosa erfolgen muß, genügt die Schnittstellenbeschreibung in *IDL*, um Dienste entfernter Objekte nutzen zu können. Bei der Entwicklung entsprechender Clientprogramme ist es nicht notwendig zu wissen, auf welche Art und in welcher Programmiersprache ein CORBA-konformer Server realisiert wurde.

5.2.2 RPC

Aus praktischen Gründen wird parallel zu CORBA eine proprietäre RPC-Implementierung in diesem Groupwaresystem eingesetzt. Der Grund liegt darin, daß eine Vielzahl von Programmen in der wissenschaftlichen Arbeitsgruppe MIPS in der Programmiersprache *perl* entwickelt werden. Es existiert derzeit jedoch kein CORBA-Produkt, das eine umfassende *perl*-Unterstützung anbietet. Alle Datenbankkomponenten wurden in *C++* realisiert. Gruppenmitglieder müssen in *perl*-Programmen (z.B. *CGI*-Skripten) Methoden persistenter Objekte aufrufen. Daher wird ein minimaler RPC-Mechanismus eingesetzt, der *C++*-Server und *perl*-Clients unterstützt (siehe Abbildung 5.7). Die erforderlichen Funktionalitäten werden als *perl package* für *perl* Clients bzw. als *C++* Klassen für *C++* Server bereitgestellt. Die zugrundeliegende `socket`-Kommunikation ist verdeckt. Zur Laufzeit wird für den Anwendungsprogrammierer transparent ein Client gestartet, der die verteilte Kommunikation durchführt.

Die Realisierung dieses RPC-Mechanismus innerhalb einer Datenbankkomponente parallel zur CORBA-Unterstützung ist aufgrund des modularen und hier-

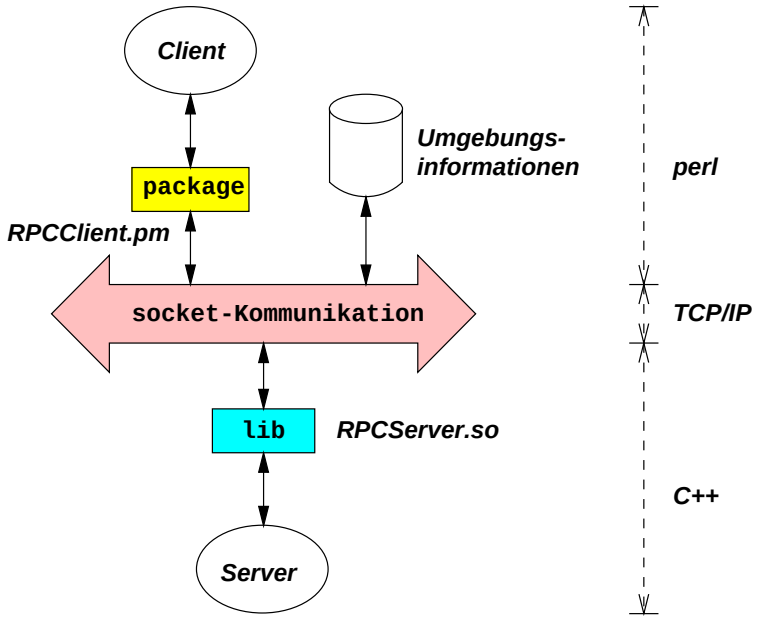

Abbildung 5.7: Ein proprietärer RPC-Mechanismus unterstützt *C++* Server und *perl* Clients.

archischen Aufbaus leicht umsetzbar. Unterstützt eine Datenbankkomponente sowohl CORBA als auch den proprietären RPC-Mechanismus, werden entsprechende Zugriffsdienste angeboten (siehe Abbildung 5.8). Diese Dienste sind spezifisch für die eingesetzte Kommunikationstechnologie. Beide Serverimplementierungen basieren auf den untergeordneten Einheiten einer Komponente: Verwaltungs- und Kontrollinstanz. Aufgrund dieser Architektur werden insbesondere Inkonsistenzen vermieden, obwohl sowohl RPC-Server als auch CORBA-Server Schnittstellen zu `update`-Methoden bereitstellen. Beide Instanzen können parallel nebeneinander existieren und über die dazwischengeschalteten Instanzen auf den gleichen persistenten Speicher zugreifen.

Zur Realisierung des RPC-Mechanismus wurde ein Protokoll entwickelt, das den Datenaustausch zwischen Client und Server realisiert. Einfache Datentypen werden diesem Protokoll folgend in entsprechende Darstellungen konvertiert und über das Netzwerk übertragen.

Im Gegensatz zu CORBA realisiert dieser RPC-Mechanismus eine direkte Kommunikation zwischen den Kommunikationspartnern ohne dazwischengeschalteten Vermittler. Die gemeinsame Laufzeitumgebung ist in einer Konfigurationsdatenbank beschrieben, zu der Clients und Server Zugriff haben. Clients können über lesenden Zugriff auf diese Datenbank den gewünschten Server loka-

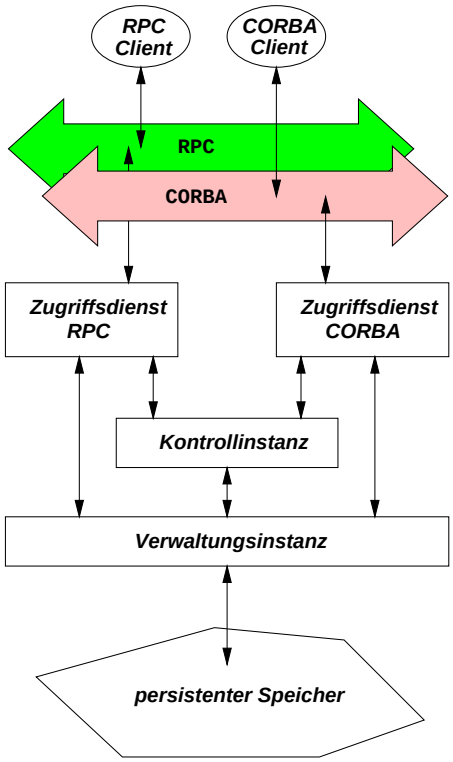

Abbildung 5.8: Der proprietäre RPC-Mechanismus und der CORBA-Standard können im Groupwaresystem parallel existieren. Je nach unterstützter Kommunikationstechnologie müssen entsprechende Zugriffsdienste angeboten werden.

lisieren.

Diese RPC-Lösung, wie sie im Prototypen realisiert wurde, ist deutlich inflexibler als die Anwendung von CORBA. Es existiert keine standardisierte Schnittstellenbeschreibungssprache. Die Konfigurationsdatenbank ist nur durch den Administrator veränderbar, nicht etwa durch den Serverprozeß selbst. Eine einfache Migration des Servers auf eine andere Maschine oder die Umleitung der Kommunikation auf einen anderen *socket-port* ist daher derzeit nur manuell durchführbar. Es wurde bei der Implementierung des RPC-Mechanismus ein Kompromiß eingegangen, da mittelfristig eine Ersetzung durch CORBA-Komponenten angestrebt wird. Die entworfene Architektur und ihre konkrete programmiersprachenabhängige Umsetzung erlaubt eine einfache und schnelle Migration des RPC-Mechanismus nach CORBA.

5.3 Informationswandler

Ziel des Informationswandlers ist die automatisierte Integration von Informationen aus externen Quellen in lokale Informationsräume. Gemäß der Definition (siehe 4.6) wird im folgenden auf die Realisierung der Teile *Importeinheit*, *Aufbereitungseinheit* sowie *Intergrationseinheit* näher eingegangen.

5.3.1 Importeinheit

Aufgabe der Importeinheit ist der Zugriff auf Daten externer Quellen und deren Ablage in einer internen Struktur, über die die Aufbereitungseinheit zugreifen kann. Durch die Importeinheit erfolgt eine Datenkonvertierung auf rein syntaktischer Ebene.

Damit eine Importeinheit Daten extrahieren kann, muß die Grammatik der Sprache, in der die Quelle verfaßt ist, beschrieben werden. Durch Parser kann auf syntaktischer Ebene festgestellt werden, ob ein gegebenes Wort (hier externe Daten) ein Wort der durch die Grammatik beschriebenen Sprache ist. Zur Realisierung können Parsergeneratoren eingesetzt werden, die anhand spezifizierter Grammatiken entsprechende Parser automatisiert erzeugen.

Werden standardisierte Sprachen eingesetzt, kann auf existierende Parser zurückgegriffen werden. Ein Beispiel ist CORBA: werden Daten durch CORBA-konforme Server angeboten, werden exportierte Schnittstellen, die einen entfernten Datenzugriff ermöglichen, in der Sprache *IDL* verfaßt. In diesem Fall besteht die Importeinheit aus einem CORBA-Client. Ein weiteres Beispiel für eine standardisierte Sprache ist XML (siehe z.B. [Lau98]).

Für den Datenaustausch zwischen Import- und Aufbereitungseinheit müssen geeignete Strukturen geschaffen werden. Diese müssen möglichst allgemein sein, um zukünftige Quellen ohne Veränderung der Aufbereitungseinheit integrieren zu können. Werden von den realisierten Parsern Datenstrukturen im Hauptspeicher angelegt, muß die nachgeschaltete Aufbereitungseinheit in der Lage sein, auf diesen Speicherbereich zugreifen zu können. Es müssen Mechanismen realisiert werden, die eine Navigation durch diese interne Repräsentation erlauben, da nicht immer alle importierten Daten von der Aufbereitungseinheit benötigt werden.

Eine flexible Lösung ist die Anwendung von XML für diesen Datentransfer: Die Importeinheit produziert als Ergebnis eine XML-Datei, die von der Aufbereitungseinheit unter Einsatz existierender XML-Parser eingelesen werden kann. In diesem Fall muß keine weitere Sprache oder Datenstruktur entwickelt werden, um temporär Daten intern verwalten zu können. Der Nachteil dieser Lösung gegenüber Datenstrukturen im Hauptspeicher ist die geringere Performanz, da zusätzlich Dateien geschrieben und wieder gelesen werden müssen. Allerdings ist eine Anpassung an neue Gegebenheiten leichter umzusetzen.

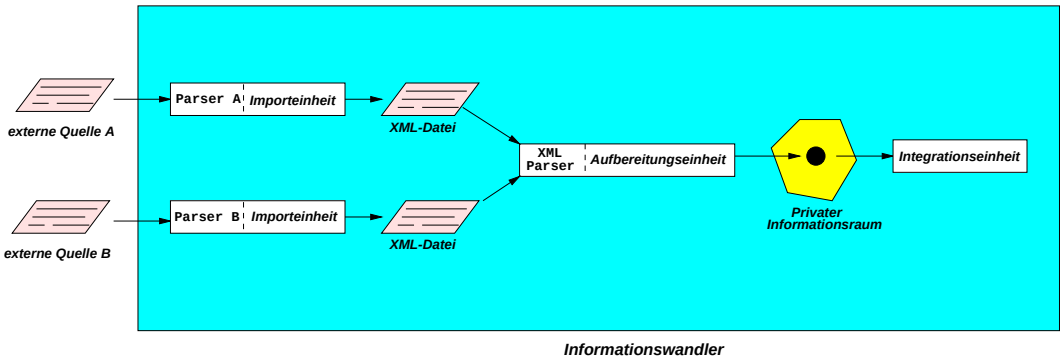

Abbildung 5.9: Anwendung von XML zum Datentransfer innerhalb eines Informationswandlers. Importeinheiten produzieren aus unterschiedlichen Quellen als Ergebnis XML-Dateien, die von der Aufbereitungseinheit unter Einsatz existierender XML-Parser eingelesen werden können. Anhand der Informationen in den XML-Dateien können entsprechende Klassen des Objektmodells instantiiert werden. Der Aufbereitungseinheit muß daher das zugrundeliegende Objektmodell bekannt sein.

Unabhängig von der eingesetzten Strategie zum Datenimport kann die Aufbereitungseinheit entwickelt werden, da die Sprache der Quelle an dieser Stelle bereits verdeckt ist.

5.3.2 Aufbereitungseinheit

Die Aufbereitungseinheit erzeugt neue Objekte im Anwendungskontext, die insbesondere eine neue eindeutige Objekt-ID enthalten. Benötigte Informationen werden aus der von der Importeinheit erzeugten Struktur extrahiert. An dieser Stelle ist Wissen um die Semantik der entsprechenden Daten und Objekte vonnöten. Es findet eine semantische Informationskonvertierung von der internen Darstellung zwischen Import- und Aufbereitungseinheit zum Objekt gemäß dem zugrundeliegenden Objektmodell der jeweiligen Anwendung statt. Diese Umsetzung erfolgt in semantischen Methoden, die manuell erzeugt werden müssen. An dieser Stelle kann außer dem Zugriff auf externe Informationen keine Rechnerunterstützung bei der Entwicklung angeboten werden.

Beispielsweise werden Autorenzeilen in Literaturzitaten von verschiedenen externen Datenbankbetreibern und Literaturdiensten unterschiedlich organisiert. Im Format der Datenbank *EMBL Nucleotide Sequence Database* wird die Autorenzeile von [MHK+99] beispielsweise folgendermaßen dargestellt:

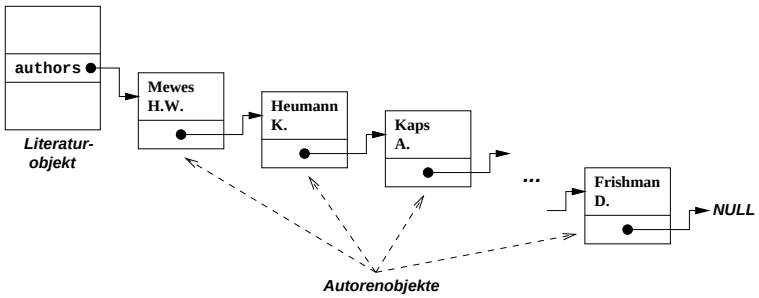

Abbildung 5.10: Objektorientierte Repräsentation von Autoren eines Literaturzitats.

```
AU      Mewes HW, Heumann K, Kaps A, Mayer K,
        Pfeiffer F, Stocker S, Frishman D
```

Bei der textuellen Repräsentation der Proteinsequenzdatenbank *PIR-International* erfolgt folgende Darstellung:

```
#authors Mewes, H.W.; Heumann, K.; Kaps, A.; Mayer,
         K.; Pfeiffer, F.; Stocker, S.; Frishman, D.
```

In der objektinternen Repräsentation wird eine Liste von Autoren abgelegt (siehe Abbildung 5.10), gemäß der Reihenfolge des Auftretens in der entsprechenden Publikation.

5.3.3 Integrationseinheit

Die Aufbereitungseinheit erzeugt neue Objekte in persönlichen Informationsräumen. Aufgabe der Integrationseinheit ist die Migration dieser Objekte in die entsprechende Datenbankkomponente, gegenüber der sie als Client auftritt. Objekte werden unter Verwendung der Kommunikationsschicht an den Server der entsprechenden Datenbankkomponente übergeben. Dazu wird ein entsprechender Dienst des Zugriffsdiensts der Datenbankkomponente in Anspruch genommen. Die weitere Prozessierung erfolgt dort. Wurde ein Objekt erfolgreich integriert, wird es aus dem persönlichen Informationsraum des Gruppenmitglieds bzw. des entsprechenden Agenten entfernt.

Geht die Antwort der Datenbankkomponente über das erfolgreiche Einfügen verloren (z.B. aufgrund eines Prozeß- oder Netzwerkausfalls), wird die durchgeführte Prozessierung zu einem späteren Zeitpunkt erkannt (siehe Abbildung 5.11). Bleibt das Resultat aus, versucht die Integrationseinheit erneut das Objekt einzufügen. Die Kontrollinstanz der Datenbankkomponente, die Redundanzfreiheit sicherstellt, erkennt diese Situation und übermittelt eine entsprechende Nachricht

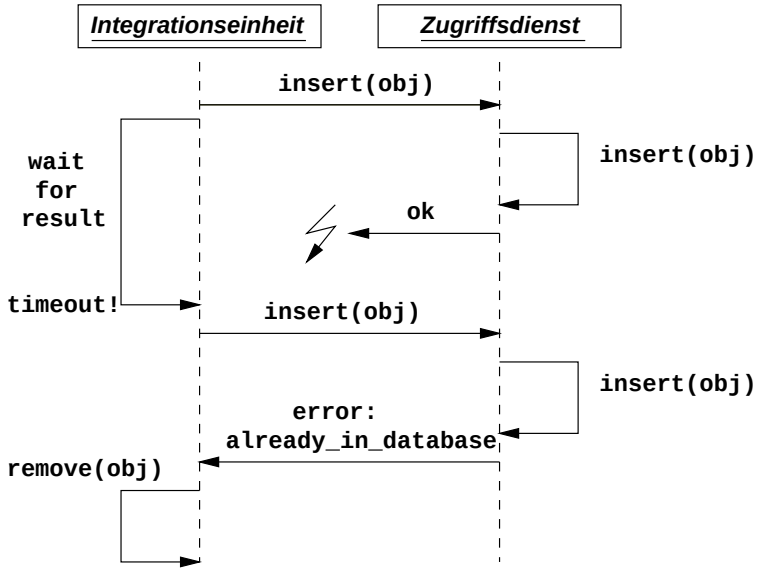

Abbildung 5.11: Sicherstellung der Redundanzfreiheit in einer verteilten Umgebung: das erfolgreiche Einfügen eines Objekts wird zu einem späteren Zeitpunkt erkannt und dem Client mitgeteilt. Eine Doppelprozessierung findet nicht statt.

an den entfernten Client. Dieser entfernt das Objekt aus dem persönlichen Informationsraum.

5.3.4 Manuelle Bearbeitungen

Sollen im Rahmen des Datenimports manuelle Korrekturen bzw. Erweiterungen durchgeführt werden, können Aufbereitungs- und Integrationseinheit voneinander gekoppelt werden. Nach Abschluß der Aufbereitungsphase liegen neue Objekte im persönlichen Informationsraum des Gruppenmitglieds vor. Nach einer entsprechenden Notifikation können manuelle Modifikationen durchgeführt werden. Sind diese abgeschlossen, wird die Integrationseinheit angestoßen. Dazu wird Gruppenmitgliedern eine Bedienoberfläche angeboten, die das Bearbeiten von Objekten sowie das Anstoßen der Integrationseinheit durch einfachen Mausklick ermöglicht.

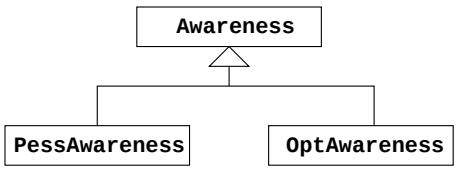

Abbildung 5.12: Modellierung von Awareness-Informationen in einer Klassenhierarchie. Spezielle Anforderungen an Awareness-Informationen werden in weiteren Klassen abgebildet, die von diesen Awareness-Basisklassen abgeleitet sind.

5.4 Awareness

Awareness wurde in 4.7 als Wissen definiert, auf dem das Verständnis des Arbeitgeschehens beruht. Um dieses Wissen zu erzeugen, sind Informationen über das Geschehen in den Gruppen-Informationsräumen vonnöten, die Gruppenmitgliedern zugänglich sein müssen. Dies kann aktiv durch das Groupwaresystem erfolgen oder passiv durch entsprechende Anfragen der Gruppenmitglieder.

Folgende Fragen, die gesamte Gruppen-Informationsräume betreffen, werden von Gruppenmitgliedern gestellt:

- *Gibt es neue Objekte im Informationsraum, die ich noch nicht gesehen habe?*
- *Fehlen Objekte im Informationsraum, d.h. ist der Informationsraum unvollständig?*

Neben diesen Awareness-Informationen über gesamte Informationsräume existieren zu einzelnen Objekten entsprechende Daten. Zur Modellierung dieser objektspezifischen Awareness wurden Awareness-Klassen eingeführt. Die Basisklasse `Awareness` enthält Informationen, die generell von Interesse sind: Zeitstempel und Name des Gruppenmitglieds, das das Objekt erzeugt bzw. zuletzt modifiziert hat. Darüberhinaus wird die Möglichkeit angeboten, freie Annotationen einzutragen.

In der nächsten Ebene sind Awareness-Klassen enthalten, die je nach eingesetzter Strategie zur Nebenläufigkeitskontrolle weitere Informationen verwalten (siehe Abbildung 5.12). Die Klasse `PessAwareness` enthält zusätzliche Informationen über den Zeitpunkt der Anforderung des assoziierten Anwendungsobjekts sowie den Namen des Gruppenmitglieds bei pessimistischen Strategien. Die Klasse `OptAwareness` ist lediglich aus Entwurfsgründen eingeführt worden. Sie enthält gegenüber der Basisklasse `Awareness` keine weiteren Klassenvariablen bzw. Methoden. Anwendungsspezifische Awareness-Informationen bei Ein-

satz der optimistischen Variante zur Nebenläufigkeitskontrolle werden von dieser Klasse abgeleitet (siehe z.B. 5.5.1). Dadurch wird die Übersichtlichkeit der Awareness-Klassenhierarchie verbessert.

Die Verwaltung der Awareness-Informationen erfolgt durch Aggregation von Awareness- und Anwendungsobjekten. Awareness-Objekte erlauben die Beantwortung folgender Fragen:

- *Wurde ein bestimmtes Informationsobjekt seit einem gewissen Datum modifiziert?*
- *Wenn ich ein Informationsobjekt nicht bearbeiten kann: von welchem Gruppenmitglied wird dieses Objekt seit wann bearbeitet und wie kann ich meine Kollegin/meinen Kollegen erreichen?*

Dazu werden in Awareness-Objekten im Rahmen von Bearbeitungen entsprechende Daten eingetragen (z.B. Zeitstempel oder Informationen, wer ein Objekt bearbeitet). Über die Menge aller Awareness-Objekte können Anfragen gestellt werden. Als Ergebnis erhält das Gruppenmitglied eine Liste von Objekten, die die Anfragekriterien erfüllen. Diese Funktionalität wird von der Datenbankkomponente angeboten.

Awareness-Informationen können kombiniert werden. Aus Awareness-Objekten von Anwendungsobjekten etwa kann die Information extrahiert werden, welche Gruppenmitglieder welche Objekte bearbeiten. Aus den Informationen, die über das gesamte System bestehen, kann ermittelt werden, ob ein bestimmtes Gruppenmitglied im System angemeldet ist und ggf. an welchem Rechner. Durch Kombinieren dieser Informationen kann ein Gruppenmitglied entscheiden, mit welchem und auf welche Art es mit seinem Kollegen/seiner Kollegin in Kontakt treten kann (direkt, Telefon, Email, etc.).

5.5 Ausgewählte Applikationen

Nach der Darstellung von Implementierungsaspekten allgemeiner Teile des Groupwaresystems wird im verbleibenden Teil dieses Kapitels auf konkrete Applikationen eingegangen, die prototypisch implementiert und in der Arbeitsgruppe eingesetzt wurden.

5.5.1 Literaturverwaltung

Ziel der zentralen Literaturverwaltung ist, einen fokussierten und redundanzfreien Datenbestand an Literaturzitaten unterschiedlichen Typs zu administrieren. Die Realisierung erfolgt als Datenbankkomponente gemäß der Definition in 4.3.2.

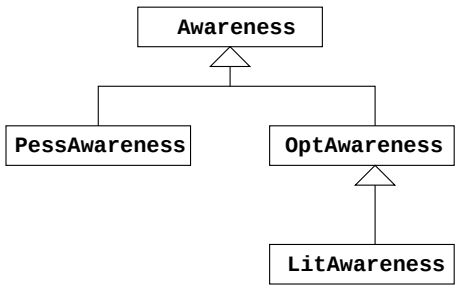

Abbildung 5.13: Für die Literaturverwaltung existiert eine eigene Awareness-Klasse `LitAwareness`, die in die allgemeine Awareness-Klassenhierarchie eingebunden ist.

Eine objektorientierte Datenbank, die vom zugrundeliegenden OODBMS *(ObjectStore)* verwaltet wird, realisiert den persistenten Speicher. Das assoziierte Anwendungsobjektmodell ist eine Klassenhierarchie bestehend aus zwei Ebenen: alle konkreten Literaturtypen werden als individuelle Klassen realisiert, die von einer allgemeinen Basisklasse abgeleitet werden (siehe Abbildung 4.10). Die Basisklasse enthält neben den allgemeinen Teilen aller Literaturtypen (z.B. Autorenzeile) ein Awareness-Objekt, das objektspezifische Awareness-Informationen enthält.

Optimistische Nebenläufigkeitskontrolle

Als Strategie zur Nebenläufigkeitskontrolle wird die optimistische Variante eingesetzt (vgl. 4.2.2). Es wird für die Literaturverwaltung eine eigene Awareness-Klasse `LitAwareness` in die Hierarchie eingefügt (siehe 5.4), die von der Klasse `OptAwareness` abgeleitet ist (siehe Abbildung 5.13). Dementsprechend enthält das Awareness-Objekt einen Zeitstempel, der den Zeitpunkt der letzten Modifikation am Zustand des Objekts festhält. Ferner existiert ein freies Annotationsfeld, in dem Gruppenmitglieder beliebige Bemerkungen zu diesem Objekt eintragen können. Diese Bemerkungen können von allen Gruppenmitgliedern eingesehen und verändert bzw. ergänzt werden. Schließlich wird die Information verwaltet, aus welcher Quelle dieses Zitat stammt.

Redundanzfreiheit

Zur Sicherstellung der Redundanzfreiheit muß in der Kontrollinstanz entschieden werden, ob ein bestimmtes Zitat bereits in der Datenbank enthalten ist. Die Durchführung dieses Tests ist abhängig vom Typ des Zitats. Anhand zweier Beispiele soll dieser Test veranschaulicht werden.

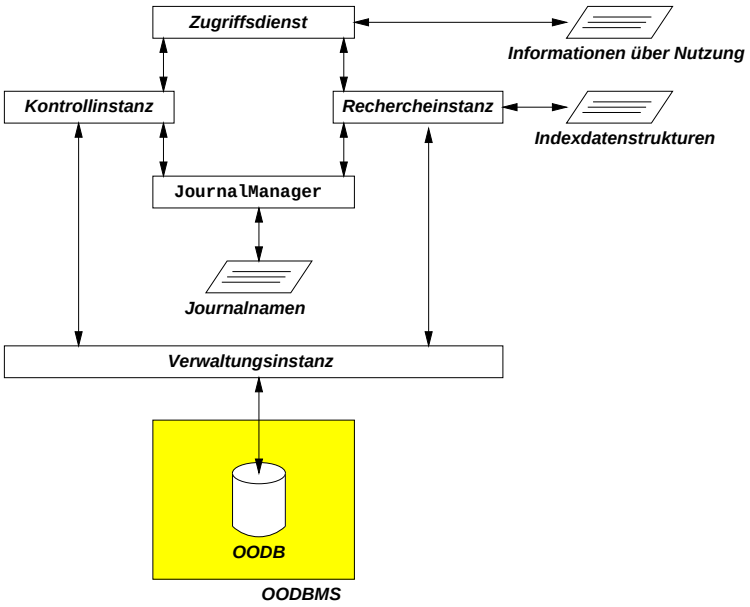

Abbildung 5.14: Sicherstellung der Redundanzfreiheit, speziell bei wissenschaftlichen Artikeln, durch Realisierung eines `JournalManagers`.

Artikel Bei *Journalzitaten* sind die entscheidenden Kriterien der Journalname, die Ausgabe, die Seitennummer der Seite, auf der der Artikel beginnt, sowie der Erstautor. Anhand dieser Angaben wird durch eine Anfrage an die Literaturverwaltung ermittelt, ob dieser spezifizierte wissenschaftliche Aufsatz bereits im Datensatz enthalten ist.

Potentiell neue Zitate werden aus verschiedenen externen Quellen extrahiert, die unterschiedliche Abkürzungen für wissenschaftliche Journale verwenden. Daher muß ein Mechanismus realisiert werden, der trotz unterschiedlicher Varianten in der Schreibweise bzw. unterschiedlicher Abkürzungen eines Journals die inhaltliche Redundanzfreiheit gewährleistet.

Dieser Mechanismus ist so realisiert, daß innerhalb der Literaturverwaltung nur eine bestimmte Schreibweise bzw. Abkürzung eines Zeitschriftennamens verwendet wird. Zusätzlich existiert eine Liste mit Alternativen zu dieser Schreibweise. Je nach Dienst wird der übergebene Journalname anhand dieser Liste auf die in der Literaturverwaltung verwendete Schreibweise abgebildet. Dies geschieht in der Kontrollinstanz der Datenbankkomponente durch einen `JournalManager` (siehe Abbildung 5.14). Dieser führt die Abbildungen durch und verwaltet die Liste der verschiedenen Schreibweisen zu Journalnamen. Der `JournalManager` bietet Dienste an, die es dem Administrator erlauben, neue Einträge in die Liste vornehmen zu können. Dadurch wird erreicht, daß bereits innerhalb der Verwal-

tungsinstanz und damit insbesondere innerhalb des persistenten Speichers einheitliche Schreibweisen zur Anwendung kommen. Die Rechercheinstanz bedient sich ebenfalls des `JournalManagers`. Diese Form der Realisierung hat insofern Einfluß auf die Architektur einer Datenbankkomponente, als eine echte Schichtung von Kontroll- und Verwaltungsinstanz vorliegt: Jeder Dienst wird zunächst von der Kontrollinstanz bearbeitet, die die Abbildung der Journalnamen durchführt. Anschließend wird die dadurch ggf. modifizierte Anfrage an die Verwaltungsinstanz weitergereicht.

Einreichung Bei Zitaten, die *direkte Einreichungen (submissions)* betreffen, sind die Datenbank, bei der etwas eingereicht wurde, Monat und Jahr der Einreichung sowie die Autoren die entscheidenden Kriterien für den Test zur Wahrung der Redundanzfreiheit. Haben die gleichen Autoren innerhalb eines Monats bei einer Datenbank mehrere Daten eingereicht, so wird dies in der Literaturverwaltung als ein Zitat abgebildet. Dieser unwahrscheinliche Fall wird von der *community* als annehmbar angesehen.

Die oben beschriebene echte Schichtung der Instanzen innerhalb der Datenbankkomponente hat für den Literaturtyp der Einreichung zur Folge, daß innerhalb der Kontrollinstanz keine über die Konsistenzsicherung und Gewährung der Redundanzfreiheit hinausgehende Bearbeitungen durchgeführt werden. Entsprechende Anfragen werden unverändert an die Verwaltungsinstanz durchgereicht.

Eingabe neuer Literaturstellen

Die automatisierte Eingabe neuer Literaturstellen wird durch Informationswandler durchgeführt (siehe 5.3). Für die Datenbankkomponente der Literaturverwaltung werden entsprechende Aufbereitungs- und Integrationseinheiten realisiert. Je nach externer Quelle müssen spezialisierte Importeinheiten entwickelt werden.

Die manuelle Eingabe neuer Sequenzen wird über HTML-Formulare ermöglicht. Nach Auswahl des gewünschten Litertaturtyps wird ein HTML-Formular erzeugt, das abhängig vom ausgewählten Typ entsprechende Eingabemöglichkeiten anbietet. Ein Einfügeknopf stößt die Bearbeitung an, bei der zunächst die eingegebenen Daten ermittelt, ein neues Literaturobjekt erzeugt und an die Datenbankkomponente übergeben wird. Das Resultat der Einfügeoperation (u.a. die für dieses Zitat vergebene Objekt-ID) wird als HTML-Dokument dem Gruppenmitglied unverzüglich präsentiert.

Werden von Gruppenmitgliedern Recherchen bei externen Literaturdatenbanken (z.B. MEDLINE) durchgeführt, sollen interessante Ergebnisse leicht in den lokalen Datenbestand übernommen werden können. Diese Recherchen erfolgen im *WWW*, d.h. entsprechende Ergebnisse liegen als HTML-Dokumente vor, die

im Browser des Gruppenmitglieds angezeigt werden. Quellen sind in diesen Darstellungen oftmals kompakt angegeben. So werden z.B. bei dem öffentlichen Teil von MEDLINE (PubMed) bei Aufsätzen der Name des Journals, die Ausgabe, Seitenzahlen und Jahr in einer einzigen Zeile angezeigt. Gruppenmitgliedern muß ermöglicht werden, diese Zeile durch einfaches *copy/paste* mit der Maus in ein entsprechendes Formular und, nach Betätigen eines Knopfs, in die zentrale Literaturverwaltung eintragen zu können. Wurde die Zeile in das HTML-Formular kopiert, wird nach Anstoß der Einfügeoperation zunächst die Eingabe entsprechend der Formatbeschreibung des externen Literaturdienstbetreibers in die literaturtypspezifischen semantischen Einheiten zerlegt. Anschließend wird ein neues Literaturobjekt gemäß den extrahierten Informationen erzeugt und an die Datenbankkomponente übergeben. Das Ergebnis dieser Operation wird dem Gruppenmitglied präsentiert.

Pflege von Literaturstellen

Im Rahmen der Pflege von Literaturstellen soll Gruppenmitgliedern ermöglicht werden, ergänzende Informationen aufnehmen bzw. Fehlerkorrekturen durchführen zu können. Die Realisierung erfolgte ebenfalls unter der Verwendung von HTML-Formularen. In der Anzeige einer Literaturstelle kann durch einfachen Mausklick das literaturtypspezifische Korrekturformular erzeugt werden. Nach Abschluß der Modifikationen (z.B. Eingabe der Zusammenfassung zu einem wissenschaftlichen Artikel) wird das modifizierte Objekt an die Datenbankkomponente übergeben.

5.5.2 Management von Proteinsequenzen

Das Management von Proteinsequenzen umfaßt das Einfügen neuer sowie die Pflege existierender Proteinsequenzen. Der wissenschaftlichen Öffentlichkeit müssen diese Informationen präsentiert werden.

Archiv für Proteinsequenzen

Gemäß den Anforderungen unter 3.2.2 wird ein Archiv für Proteinsequenzen benötigt, in das jede neue Proteinsequenz des gemeinsamen Informationsraums abgelegt wird.

Dieses Archiv wurde im vorangegangenen Kapitel als Datenbankkomponente entworfen. Die Realisierung erfolgt analog der Literaturverwaltung: als persistenter Speicher dient eine objektorientierte Datenbank, die von der Verwaltungsinstanz administriert wird. Aufgrund der geringen zu erwartenden Modifikationen existierender Objekte und der geringen Forderung nach Verfügbarkeit,

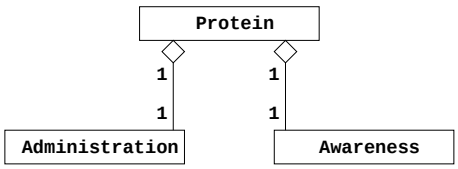

Abbildung 5.15: Ein Objekt der Annotationsdatenbank ist eine Instanz der Klasse `Protein`, das zwei weitere Instanzen der Klassen `Administration` und `Awareness` enthält.

wird von der Kontrollinstanz die Variante der pessimistischen Nebenläufigkeitskontrolle realisiert. Die Umsetzung der pessimistischen Nebenläufigkeitskontrolle erfolgt gemäß der Darstellung unter 5.1.3. Darüberhinaus sichert die Kontrollinstanz Redundanzfreiheit zu. Hierbei muß sichergestellt sein, daß die gleiche Sequenz aus der gleichen Quelle nicht doppelt im Informationsraum enthalten ist. Wird eine Sequenz aus unterschiedlichen Quellen extrahiert, werden zwei Objekte im Archiv angelegt. Der `insert`-Dienst der Datenbankkomponente des Proteinsequenzarchivs erwartet als Parameter neben der Sequenz u.a. auch die Angabe der zugrundeliegenden Quelle.

Annotationsdatenbank

Die Proteinsequenzdatenbank enthält zu einer kanonischen Proteinsequenz eine Menge biologischer Zusatzinformationen (Annotationen). Objekte des zugrundeliegenden Objektmodells werden in einer weiteren zusätzlichen Datenbankkomponente, der Annotationsdatenbank, verwaltet. Literaturzitate sind in diesen Annotationen lediglich als Verweise auf Objekte in der Literaturverwaltung enthalten. Die in der kanonischen Sequenz enthaltenen ursprünglichen Rohsequenzen können über das Sequenzarchiv erreicht werden.

Als persistenter Speicher für die Annotationsdatenbank dient eine objektorientierte Datenbank, die unter *ObjectStore* verwaltet wird. Die Kontrollinstanz implementiert eine pessimistische Variante zur Nebenläufigkeitskontrolle. Redundanzfreiheit wird über Objekt-IDs realisiert: eine Objekt-ID darf nur genau einmal im Informationsraum enthalten sein. Bei der Erzeugung eines neuen Objekts wird immer eine neue eindeutige Objekt-ID generiert.

Neben biologischen Inhalten enthält ein Objekt der Annotationsdatenbank, im folgenden Proteinobjekt genannt, aggregiert zwei weitere Objekte (siehe Abbildung 5.15):

- ein *Administrationsobjekt* aus der Sicht der Annotationsgruppe sowie
- ein *Awareness-Objekt*.

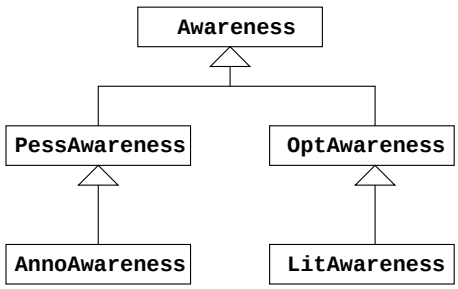

Abbildung 5.16: Für die Annotationsdatenbank existiert eine eigene Awareness-Klasse `AnnoAwareness`, die in die allgemeine Awareness-Klassenhierarchie eingebunden ist.

Das Administrationsobjekt beinhaltet Informationen, die den Prozessierungsstatus des Proteinobjekts beschreiben. Die Proteinobjekte der Annotationsdatenbank unterliegen der ständigen Pflege durch das wissenschaftliche Personal. Modifizierende Bearbeitungen können als Folge den Prozessierungsstatus des Objekts beeinflußen. Werden z.B. zwei Proteinobjekte zu einem zusammengefaßt, erhält eines der beiden Objekte den neuen Status *merged*. Dieses Objekt ist daraufhin nach außen nicht mehr sichtbar, wird jedoch nicht aus dem gemeinsamen Informationsraum entfernt.

Das Awareness-Objekt wird gemäß der Beschreibung in 5.4 realisiert. Da bei dieser Datenbankkomponente eine pessimistische Variante zur Nebenläufigkeitskontrolle eingesetzt wird, wird eine Klasse `AnnotationAwareness` von der Klasse `PessAwareness` abgeleitet (siehe Abbildung 5.16).

Der Zusammenschluß aus den drei Datenbankkomponenten Literaturverwaltung, Sequenzarchiv und Annotationsdatenbank realisiert die logische Proteinsequenzdatenbank (siehe Abbildung 5.17). Ein Proteinobjekt kann aus dem Zusammenschluß ursprünglich mehrerer verschiedener Proteinobjekte hervorgegangen sein. Wurde z.B. zunächst ein Fragment eines Proteins in die Datensammlung aufgenommen und zu einem späteren Zeitpunkt die komplette Proteinsequenz, liegen zunächst zwei Proteinobjekte vor. Im Zuge der Pflege werden diese beiden Proteinobjekte zu einem zusammengefaßt. Die Geschichte dieses Objekts bleibt jedoch nachvollziehbar: es wird auf die entsprechenden Objekte im Proteinsequenzarchiv verwiesen. Wie in Abbildung 5.17 dargestellt, enthält ein Proteinobjekt somit genau ein Adminstrations- sowie Awareness-Objekt und verweist auf mindestens ein Literaturobjekt in der Literaturverwaltung und auf mindestens ein Sequenzobjekt im Proteinsequenzarchiv.

Wie unter 4.8.3 dargestellt, beinhaltet Annotation die Aspekte *Eingabe* neu-

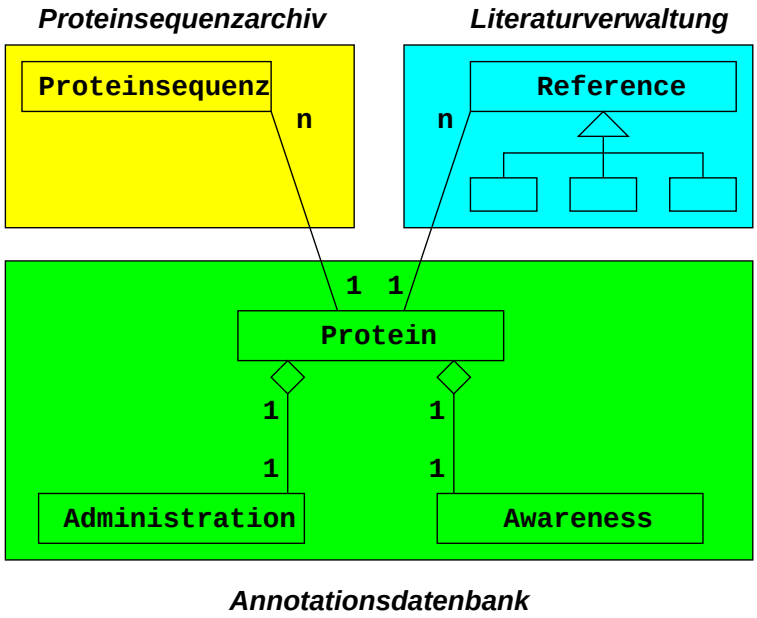

Abbildung 5.17: Der Zusammenschluß der drei Datenbankkomponenten Literaturverwaltung, Proteinsequenzarchiv sowie Annotationsdatenbank realisiert die Proteinsequenzdatenbank.

er Proteinsequenzen, die *Pflege* bereits im Informationsraum enthaltener biologischer Informationen sowie deren *Präsentation* der wissenschaftlichen Öffentlichkeit. Auf diese Teilbereiche wird im folgenden näher eingegangen.

Eingabe Neue Proteinsequenzen werden aus externen Quellen extrahiert. Zur Realisierung werden entsprechende Informationswandler eingesetzt (vgl. 4.6). Die Importeinheit muß für jede externe Quelle individuell realisiert werden. Aufbereitungs- und Integrationseinheit können unabhängig von der Quelle für die Annotationsdatenbankkomponente entwickelt werden.

Der realisierte Prototyp enthält einen Informationswandler für die Nukleinsäuredatenbank *EMBL Nucleotide Sequence Database* (siehe 2.4.1). Um die Semantiken dieser wichtigen Informationsressource adäquat behandeln zu können, enthält dieser Informationswandler eine manuelle Komponente, in der proteinspezifische Modifikationen durchgeführt werden können (siehe Abbildung 5.18). Desweiteren werden systematische Anpassungen an eine datenbankweit konsistente Nomenklatur umgesetzt. Dazu gehören z.B. Abgleiche gegen gruppeninterne Taxonomielisten, die nicht standardisiert sind und daher von unterschiedlichen Gruppen verschieden verwendet werden. Ein weiteres Beispiel ist die funktionelle

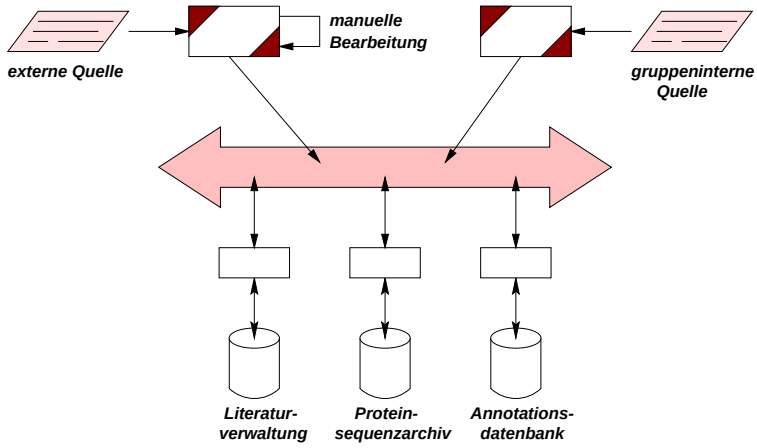

Abbildung 5.18: Die Eingabe neuer Proteinsequenzen erfolgt durch den Einsatz von Informationswandlern, die über manuelle Bearbeitungsteile verfügen können. Extrahierte Informationen werden gemäß den Objektmodellen des Groupwaresystems in Datenbankkomponenten übernommen.

Klassifikation eines Proteins gemäß eines hierarchisch geordneten Funktionskatalogs, der von der wissenschaftlichen Arbeitsgruppe MIPS entwickelt wurde und ständig angepaßt und erweitert wird (siehe Abbildung 5.19).

Aus der Sicht der Gruppenmitglieder wurde ein System geschaffen (*Protein Integration and Annotation, PrIAn*, [MFG+00]), das die Eingabe neuer Proteinsequenzen aus der Nukleinsäuredatenbank ermöglicht. Der Einsatz eines Informationswandlers, beschrieben durch seine Einheiten, ist den Gruppenmitgliedern verdeckt.

PrIAn verfügt über eine Liste von Einträgen der Nukleinsäuredatenbank, die in den gemeinsamen Informationsraum der Proteinsequenzdatenbank eingefügt werden sollen. Diese Liste wird automatisiert durch Vergleiche der Datenbestände beider Datensammlungen erstellt. Der Gruppenleiter der Annotationsgruppe kann dabei Prioritäten festlegen, um z.B. Einträge eines bestimmten Organismus bevorzugt bearbeiten zu lassen. Ein Gruppenmitglied kann nach Anmeldung den nächsten auf der Liste stehenden Eintrag bearbeiten. Alternativ wird für jedes Gruppenmitglied eine individuelle Liste von zu bearbeitenden Einträgen verwaltet. Diese Listen werden ebenfalls vom Gruppenleiter unter Beachtung spezieller Anforderungen (z.B. gleichmäßige Auslastung aller Gruppenmitglieder) bzw. unter Ausnutzung existierenden Fachwissens seitens der Gruppemitglieder erzeugt.

Wurde ein Eintrag von Gruppenmitgliedern ausgewählt, erfolgt die Darstellung des neuen Proteinobjekts nach Abschluß der Aufbereitungseinheit des Infor-

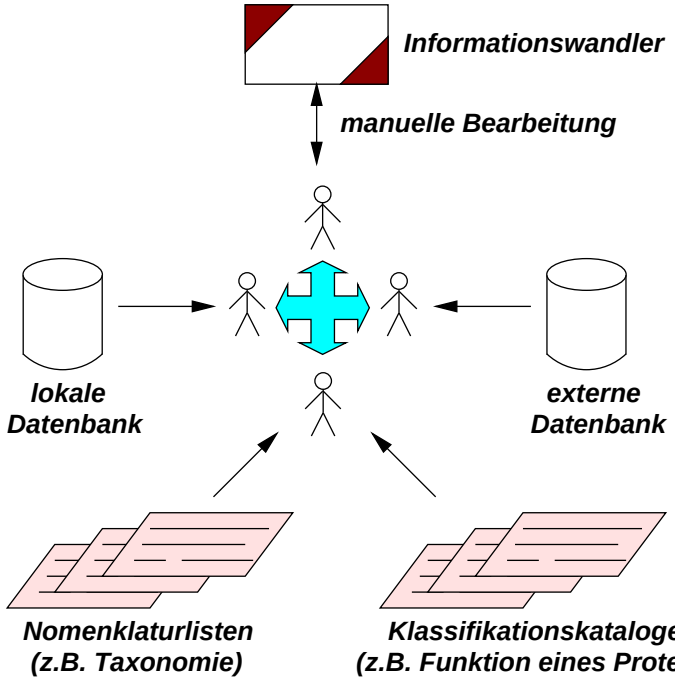

Abbildung 5.19: Die manuelle Annotation innerhalb der Gruppe erfordert Zugriffe auf verschiedene lokale oder externe Datenbanken sowie auf Listen und Kataloge, um konsistente Zusatzinformationen zu biologischen Rohdaten zu erreichen.

mationswandlers. Es ist die manuelle Bearbeitung dieses neuen Objekts möglich. Zur Unterstützung dieser Tätigkeit wird der Ausgangseintrag der Nukleinsäuredatenbank auf Anfrage angezeigt. Die im Objekt enthaltenen Informationen werden abgeglichen gegen interne Informationsrepräsentationen: Literaturzitate werden in die zentrale Literaturverwaltung übernommen und neue Proteinsequenzen werden ins Proteinsequenzarchiv eingetragen.

Die einem Eintrag einer Nukleinsäuredatenbank zugrundeliegende Sequenz kann potentiell mehrere Gene enthalten, die Proteine kodieren. Daher können als Folge der Bearbeitung eines Nukleinsäuredatenbankeintrags eine Menge neuer Proteinobjekte entstanden sein. Diese werden zunächst von der Aufbereitungseinheit des Informationswandlers im privaten Informationsraum des entsprechenden Gruppenmitglieds erzeugt. Während der manuellen Phase können Gruppenmitglieder Informationen bestimmen, die in allen neuen Proteinobjekten enthalten sein sollen (z.B. organismusspezifische Informationen). Diese Daten werden automatisiert in alle Proteinobjekte übernommen. Anschließend können alle erzeugten Proteinobjekte individuell bearbeitet werden, um genspezifische Annotationen

durchführen zu können.

Ist die manuelle Bearbeitung abgeschlossen, werden durch die Integrationseinheit des Informationswandlers alle neu erzeugten und manuell bearbeiteten Proteinobjekte in die Annotationsdatenbankkomponente migriert. Der Zustand aller Proteinobjekte wird in ihren Administrationsobjekten auf `new` gesetzt, da sie neu in den Informationsraum eingetragen wurden.

Die graphische Bedienoberfläche wurde in HTML realisiert. Eingaben werden über HTML-Formulare ermöglicht. Prozesse werden über den CGI-Mechanismus angestoßen.

Pflege Die in der Annotationsdatenbank enthaltenen biologischen Informationen müssen ständig an den aktuellen Wissensstand angepaßt werden. Diese Pflege kann automatisiert erfolgen, falls systematische Anpassungen erforderlich sind. In diesem Fall werden die durchzuführenden Änderungen in einer speziellen Sprache beschrieben und von einem Prozeß abgearbeitet. Manuelle Pflege erfolgt durch geschultes wissenschaftliches Personal, das die Annotationsgruppe darstellt.

Unabhängig, ob eine automatisierte oder manuelle Pflege durchgeführt wird, muß Konsistenz der Inhalte des gemeinsamen Informationsraums vor dem Hintergrund asynchroner Bearbeitungen durch Gruppenmitglieder und automatisiert ablaufender Prozesse gewährleistet werden. Diese Aufgabe übernimmt die Kontrollinstanz der Annotationsdatenbankkomponente. Bei dieser Datenbankkomponente wird, analog dem Archiv für Proteinsequenzen, eine pessimistische Strategie zur Nebenläufigkeitskontrolle eingesetzt.

Im realisierten Prototypen werden in privaten Informationsräumen der Gruppenmitglieder durch den *checkout*-Dienst der Annotationsdatenbankkomponente Proteinobjekte als Textdateien angelegt, d.h. es liegt eine textuelle Repräsentation der biologischen Inhalte vor. Bearbeitungen können daher mit einfachen Texteditoren durchgeführt werden. Dadurch wird die Unterstützung verschiedener Systeme und Betriebssysteme erleichtert, da diese Art von Programmen auf allen Systemen vorhanden sind. Darüberhinaus können Gruppenmitglieder diese Textrepräsentationen über Modemverbindungen auf heimische Rechner laden bzw. auf Diskette kopieren. Der Nachteil dieser Methode liegt darin, daß Gruppenmitglieder in der textuellen Repräsentation, der eine standardisierte Sprache zugrundeliegt, editieren müssen. Syntaktische Fehler (z.B. fehlende Klammern) führen zu Problemen beim abschließenden Einfügen der Proteinobjekte in die Datenbank. Angestrebt wird ein Annotationseditor, der in der plattformneutralen Programmiersprache *Java* unter Einsatz graphischer Bibliotheken (z.B. JFC *Swing*) entwickelt wurde. Voraussetzung auf dem eingesetzten Rechner ist damit lediglich das Vorhandensein der virtuellen Java Maschine und der entsprechenden Java Bibliotheken. Alternativ kann der Annotationseditor als Applet in der Browser-

software des Gruppenmitglieds ablaufen.

Wurden Bearbeitungen an einem Proteinobjekt durchgeführt, wird das Replikat unter Anwendung des Dienstes `checkin` wieder an die Datenbankkomponente übergeben. Im Prototyp wird auf Clientseite aus der textuellen Proteinbeschreibung sowie den assoziierten Awareness- und Administrations-Informationen ein neues transientes Proteinobjekt erzeugt, das über die Kommunikationsschicht an den Zugriffsdienst der Datenbankkomponente weitergereicht wird. Die Kontrollinstanz führt anhand der mitgelieferten Verwaltungsinformationen, die im Awareness-Objekt enthalten sind, Konsistenzüberprüfungen durch. Im Erfolgsfall wird das persistente Objekt in der objektorientierten Datenbank durch das modifizierte transiente Proteinobjektreplikat ersetzt. Der Prototyp der Annotationsdatenbankkomponente enthält kein Versionenmanagement. Lediglich die letzte Fassung eines Proteinobjekts ist im Datensatz enthalten.

Präsentation Biologische Inhalte, die durch Proteinobjekte der Annotationsdatenbank sowie ihren assoziierten Objekten in weiteren Datenbankkomponenten repräsentiert sind, müssen der Gruppe sowie der wissenschaftlichen Öffentlichkeit aufbereitet präsentiert werden.

Anders als etwa bei systematischen Genomsequenzierungsprojekten sind alle in der Proteinsequenzdatenbank enthaltenen Inhalte öffentlich zugänglich. Es müssen keine Sicherheitsaspekte hinsichtlich der Sichtbarkeit und des Zugriffs beachtet werden. Die einzige Einschränkung besteht darin, daß lediglich Proteinobjekte, die einen bestimmten Bearbeitungsstatus besitzen (`alive` oder `new`) angezeigt werden sollen. Alle übrigen Objekte befinden sich in Bearbeitungszuständen, die für die Pflege der Daten relevant, aber nicht für die Öffentlichkeit bestimmt sind. Da der Bearbeitungszustand eines Proteinobjekts im Adminstrationsobjekt enthalten ist, kann zu jedem Objekt die Sichtbarkeit leicht ermittelt werden. Dazu wird eine entsprechende Anfrage an das Administrationsobjekt eines Proteinobjekts gestellt.

Die Art der Darstellung biologischer Inhalte kann in unterschiedlichen Formen erfolgen. Darüberhinaus kann zwischen statischer und dynamischer Präsentation unterschieden werden. Grundsätzlich verfügt jedes Objekt des zugrundeliegenden Objektmodells über entsprechende Präsentationsmethoden, die klassenspezifisch implementiert wurden. Alle Klassen eines Klassenbaums verfügen über Methoden gleichen Namens, um eine bestimmte Präsentation durchführen zu können. Dadurch kann zur Laufzeit die klassenspezifische Methode individueller Objekte ausgenutzt werden *(dynamic binding)*.

Im realisierten Prototyp erfolgt unter Anwendung dieser Methodik eine Darstellung im Format `CO2`, einem standardisierten Format, das im Rahmen des Projekts *PIR-International* entwickelt wurde. Diese Darstellung kann sowohl für sta-

tische als auch für dynamische Präsentationen eingesetzt werden.

Bei der statischen Variante werden zu einem bestimmten Zeitpunkt alle Proteinobjekte gemäß dieser Darstellung ausgegeben und gesichert. Dieser so repräsentierte Datensatz kann via `ftp` oder Massenmedium (z.B. CD-ROM) vertrieben werden. Dadurch wird es interessierten Wissenschaftlern ermöglicht, eine lokale Kopie der enthaltenen biologischen Informationen zu erhalten. Aufgrund der definierten Sprache, in der die Informationen abgefaßt wurden, ist eine maschinenorientierte Weiterverarbeitung möglich.

Bei der dynamischen Variante wird auf das derzeit in der Datenbankkomponente enthaltene persistente Objekt zugegriffen. Die Aktualität der Informationen letzterer Methode ist höher. Bei schlechten Netzwerkverbindungen kann die Antwortzeit jedoch deutlich länger sein, als beim Vorliegen einer lokalen Kopie des Datensatzes.

Darüberhinaus wurde zur Unterstützung der dynamischen Präsentation ein Prototyp entwickelt, der unter Anwendung des *Model-View-Controller* Entwurfmusters *(Pattern)* realisiert wurde (siehe z.B. [GHJV95]). Das Modell ist die Proteinsequenzdatenbank, realisiert als Menge von Datenbankkomponenten, die Kontrolle besteht aus einer Java-Applikation. Die Darstellung erfolgt, nach Inhalten organisiert, in Fenstern auf dem Desktop des Gruppenmitglieds: ein Fenster dient der Formulierung von Anfragen an die Datenbank, ein Fenster stellt Proteinobjekte, ein weiteres Literaturobjekte dar. Kommunikation zwischen diesen Fenstern wird durch die Kontrolle ermöglicht. Wird z.B. bei der Proteindarstellung auf einen Literaturverweis geklickt, wird im Literaturfenster das entsprechende Literaturobjekt aus der zentralen Literaturverwaltung dargestellt.

Dieser Prototyp kann entsprechend erweitert werden, um graphische Darstellungen annotierter Aspekte eines Proteins (z.B. stark konservierte Bereiche, einzelne Bindungsstellen etc.) abhängig von ihrer Lokalisation auf der Aminosäuresequenz zu ermöglichen.

5.5.3 Genomsequenzierungsprojekte

Der realisierte Prototyp unterstützt Genomsequenzierungsprojekte nur insofern, als Überschneidungen zur Proteinsequenzdatenbank betroffen sind. Es wurde konkret eine Unterstützung für die Gruppe des Sequenzierungsprojekts der Pflanze *Arabidopsis thaliana* implementiert.

Haben die verwalteten Sequenzen und ihre Annotationen einen Zustand erreicht, der die Veröffentlichung erlaubt, kann von einem Gruppenmitglied ein Veröffentlichungsverfahren eingeleitet werden. Veröffentlichung bedeutet in diesem Kontext, daß die mit biologischem Wissen angereicherten Sequenzen bei der Nukleinsäuredatenbank *EMBL Nucleotide Sequence Database* sowie der Proteinsequenzdatenbank *PIR-International* eingereicht werden (siehe Abbildung 4.21).

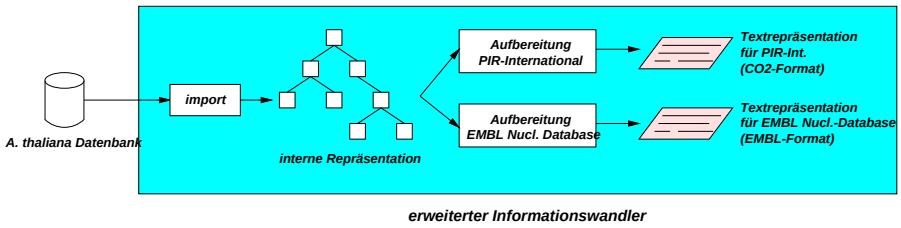

Abbildung 5.20: Ein spezialisierter Informationswandler verfügt über zwei Aufbereitungseinheiten. Dadurch ist die parallele Einreichung bei zwei verschiedenen und voneinander unabhängigen Datenbankbetreibern möglich.

Dazu müssen lokale Informationsrepräsentationen gemäß den Vorgaben dieser Datenbankbetreiber umgesetzt werden. Dazu wurde ein spezialisierter Informationswandler implementiert. Da in diesem Fall Einreichungen bei zwei Datenbankbetreibern erfolgen, verfügt dieser Informationswandler über zwei Aufbereitungs- und Intergrationseinheiten (siehe Abbildung 5.20). Nach der syntaktischen Aufbereitung durch die Importeinheit erzeugen die Aufbereitungseinheiten keine Objekte im Kontext des Groupwaresystems. Vielmehr werden Textdateien in definierten Formaten gemäß der Datenbanken, bei denen eine Einreichung erfolgen soll, erzeugt. Nachgeschaltete Integrationseinheiten führen die Einreichungen durch.

Wurden durch die Aufbereitungseinheiten entsprechende Textrepräsentationen erzeugt (im *EMBL*-Format bzw. `CO2`-Format), werden diese durch die jeweilige Integrationseinheit bei der entsprechenden Datenbank eingereicht. Im Fall der Nukleinsäuredatenbank bedeutet dies, daß die Textdateien via Email an den Datenbankbetreiber verschickt werden. Bei der Proteinsequenzdatenbank erfolgt die Migration der zuvor erzeugten Dateien in einen besonderen Informationsraum. Dort enthaltene Dateien werden vom Informationswandler der Proteinsequenzdatenbank weiterverarbeitet. Wie unter 5.5.2 beschrieben, werden diese neuen Proteinobjekte in das Annotationssystem *PrIAn* integriert. Nach Abschluß dieser Integration liegen sie als neue Proteinobjekte in der Annotationsdatenbankkomponente vor.

Die Implementierung erfolgte unter Einsatz der Programmiersprache *perl*, da viele Operationen auf Textdateien durchzuführen sind. *Perl* bietet dazu entsprechend mächtige Sprachkonstrukte an, die eine einfache Anpassung im Zuge von Formatänderungen seitens der Datenbankbetreiber erlauben.

Dieses Veröffentlichungsverfahren kann von jedem Gruppenmitglied initiiert werden. Die Koordination unter den Gruppenmitgliedern erfolgt durch mündliche Absprachen. Allerdings werden zu jedem eingeleiteten Verfahren im lokalen Gruppeninformationsraum Informationen abgelegt, die von allen Gruppenmitglie-

dern eingesehen werden können. Darin ist u.a. die Information enthalten, welches Gruppenmitglied das Verfahren wann eingeleitet hat.

Kapitel 6

Einsatz und empirische Akzeptanzanalyse des prototypischen Groupwaresystems

Der im vorangegangenen Kapitel dargestellte, prototypisch realisierte Teil des entworfenen Groupwaresystems wurde in einer wissenschaftlichen Arbeitsgruppe eingesetzt. Dieses Kapitel beschäftigt sich mit dem Einsatz sowie der anschließend durchgeführten Analyse der Akzeptanz durch die Anwender in der Gruppe MIPS. Zu diesem Zweck wurde eine Umfrage unter den Gruppenmitgliedern durchgeführt. Die erzielten Ergebnisse werden dargestellt und diskutiert.

6.1 Einsatz des prototypischen Groupwaresystems

Folgende im vorangegangenen Kapitel dargestellten Applikationen des Groupwaresystems wurden in der Arbeitsgruppe MIPS eingesetzt:

- Zentrale Literaturverwaltung (siehe 5.5.1),
- Management von Proteinsequenzen (siehe 5.5.2),
- Partielle Unterstützung systematischer Genomsequenzierungsprojekte (5.5.3).

Diese Applikationen basieren auf folgenden Basiskomponenten:

- Datenbankkomponente (siehe 5.1),
- Kommunikationsschicht (siehe 5.2),
- Informationswandler (siehe 5.3).

Der Einsatz erfolgte in der Gruppe MIPS, die am Ende der Untersuchung aus 28 Mitarbeitern bestand. 17 davon nutzten Teile des beschriebenen Sytems über einen Zeitraum von fünf Monaten bis drei Jahren (je nach Dauer der Zugehörigkeit zur Gruppe bzw. Verfügbarkeit der entsprechenden Anwendung). Der Einsatz der Anwendungen erfolgte inkrementell: als erste Applikation wurde die zentrale Literaturverwaltung bereitgestellt, bei der in ihrer mittlerweile dreijährigen Verfügbarkeitsdauer verschiedene Versionsrevisionen durchgeführt wurden. Änderungen betrafen in erster Linie das zugrundeliegende objekt-orientierte Datenbankmanagementsystem *ObjectStore*. Der Funktionsumfang wurde schrittweise erweitert. Der Entwurf aus Kapitel 4.8.4 blieb während der durchgeführten Erweiterungen unangetastet. Es folgte die partielle Unterstützung des systematischen Genomsequenzierungsprojekts *A. thaliana*. Die Applikationen zum Management von Proteinsequenzen schloßen sich an. Abschließend erfolgte die Anbindung des Genomsequenzierungsprojektes *A. thaliana* an die Proteinsequenzdatenbankgruppe. Dieser letzte Schritt erfolgte ca. sechs Monate vor Start der Akzeptanzanalyse. D.h. jede CSCW-Anwendung war mindestens ein halbes Jahr in der Gruppe im Einsatz. Die Gruppenmitglieder haben daher mindestens ein halbes Jahr mit den dargestellten Anwendungen gearbeitet.[1]

6.2 Datenerhebung

Um eine Akzeptanzanalyse durchführen zu können, die wissenschaftlichen Ansprüchen gerecht wird, muß eine geeignete Methode zur Erhebung der dafür notwendigen Daten ausgewählt und angewendet werden. In diesem Abschnitt werden theoretische Überlegungen sowie Techniken der empirischen Sozialforschung vorgestellt und auf ihre Anwendbarkeit auf den hier zugrundeliegenden Kontext hin untersucht. Die schließlich eingesetzte Methode wird beschrieben.

6.2.1 Empirische Sozialforschung

In der modernen Gesellschaft werden von unterschiedlichsten Gruppen systematisch Informationen für eine Vielzahl von Problemstellungen benötigt. In nur wenigen Fällen wird dabei auf die theoretisch und experimentell begründete Basis der empirischen Sozialforschung zurückgegriffen. Laut [SHE95] sind von einer extremen Trivialisierung in erster Linie *Umfragen* betroffen: „Die Unkenntnis über Methoden empirischer Sozialforschung führt hier zu den Absurditäten, die sich täglich in den Medien als *Ergebnis* von *Umfragen* finden und die das Bild der Sozialforschung in der Öffentlichkeit zunehmend prägen“ ([SHE95], S. 5).

[1] Mit einer Ausnahme: ein Gruppenmitglied ist erst fünf Monate vor Start der Analyse zur Gruppe gestoßen.

Die empirische Sozialforschung stellt eine Sammlung von Techniken und Methoden zur Verfügung, um eine wissenschaftliche Untersuchung über Sachverhalte in der Natur und/oder Gesellschaft durchführen zu können. Sie dient der systematischen Evaluierung von Hypothesen.

Bevor Hypothesen, die dieser Akzeptanzanalyse zugrundeliegen, formuliert werden, erfolgen zunächst allgemeine Betrachtungen zu Datenerhebungstechniken.

6.2.2 Datenerhebungstechniken

Nach [SHE95] können drei Verfahren zur Datenerhebung unterschieden werden:

- Befragung
- Beobachtung
- Inhaltsanalyse

Bei der Wahl eines geeigneten Verfahrens müssen einige Aspekte beachtet werden. Es ist z.B. zu hinterfragen, inwiefern unerwünschte Reaktionen der Untersuchungsobjekte durch die eingesetzte Erhebungstechnik hervorgerufen werden. Grundsätzlich ist die Wahl abhängig von der Angemessenheit in bezug auf das Untersuchungsziel.

Die *Befragung* kann als Standardmethode für die Erhebung angesehen werden ([SHE95], S. 299). Es wurden für die drei Formen der Befragung, mündlich, schriftlich und telefonisch, eigene Lehren entwickelt, die sich z.B. mit der handwerklichen Ausarbeitung von Fragebögen beschäftigen.

Als ursprünglichste Datenerhebungstechnik kann die *Beobachtung* angesehen werden. Im Gegensatz zur alltäglichen Beobachtung, erfolgt die wissenschaftiche Beobachtung kontrolliert und systematisch. Beobachtungsinhalte werden systematisiert ([SHE95], S. 355). Im Gegensatz zur Befragung ist bisher keine allgemeine Theorie für die Beobachtung entworfen worden. Grob können die unterschiedlichen Beobachtungsverfahren in *direkte* und *indirekte* Beobachtung klassifiziert werden. Steht bei der direkten Beobachtung das Verhalten im Mittelpunkt, erfolgt die indirekte Beobachtung anhand der Spuren und Auswirkungen des Handelns. Die technologische Entwicklung hat die traditionellen Werkzeuge, Papier und Bleistift, ergänzt um Tonbandgeräte oder Film- und Videoaufnahmen. Dadurch kann eine objektivere Analyse durch nachträgliches Vergleichen durchgeführt werden.

Bei den beiden vorgestellten Methoden, der Befragung und der Beobachtung, ist den untersuchten Objekten bewußt, daß sie Gegenstand einer Untersuchung

sind. Bei der *Inhaltsanalyse* werden Texte aller Art, sowie Rundfunk- und Fernsehsendungen untersucht. Diese Methode kann als nicht-reaktiv angesehen werden, da weder Produzenten noch Leser bzw. Zuschauer direkt durch eine Inhaltsanalyse betroffen sind ([SHE95], S. 372).

In dieser Arbeit wurde die Befragung als Datenerhebungsmethode ausgewählt, da die Vorteile, wie z.B. parallele Befragung und geringe Kosten, ausgenutzt werden konnten. Eine Beobachtung erfordert einen deutlich höhren Zeitaufwand, selbst wenn Technologien, wie Videoaufnahmen, eingesetzt würden. Die Aufzeichnungen müssen anschließend analysiert und verglichen werden. Darüberhinaus verhalten sich durch eine Videokamera beobachtete Personen meistens nicht natürlich, was zu Verzerrungen führt. Eine Inhaltsanalyse war nicht durchführbar, da in diesem Fall keine Texte im klassischen Sinn produziert werden.

Im folgenden werden allgemeine theoretische Grundlagen für die Durchführung von Befragungen dargestellt und die eingesetzte Befragungsart, die *schriftliche Befragung*, begründet.

6.2.3 Theoretische Grundlagen für Befragungen

Nach [vRHM94] kann eine Befragung in die Phasen *Vorbereitung*, *Konstruktion der Fragen*, Bestimmung der *Art der Befragung* sowie der *Auswertung und Präsentation* eingeteilt werden.

Vorbereitungsphase

Die Phase der Vorbereitung befaßt sich mit der Zielrichtung der geplanten Befragung, um Probleme bei der Auswertung der ermittelten Daten zu vermeiden. Eine Zielrichtung kann dabei umso präziser formuliert werden, je besser das Umfeld bekannt ist.

Um ungenaue Fragestellungen zu vermeiden, muß entschieden werden, was erfragt werden soll. Dabei muß beachtet werden, welche Aspekte von Interesse sind, d.h. sollen Einstellungen, Meinungen, Schätzungen oder Wissen erhoben werden. Es ist festzulegen, welche Personen oder -gruppen befragt werden sollen. Neben Privatpersonen können auch Experten, Haushalte, Unternehmen, etc. Ziel einer solchen Datenerhebung sein. Wurde der Personenkreis festgelegt, ist zu ermitteln, wie die Personen erreicht werden können. Man muß entscheiden, ob sie intellektuell und sprachlich für eine Befragung geeignet sind. In dem hier vorliegenden Fall ist die Personengruppe durch die Mitglieder der Arbeitsgruppe festgelegt. Die Erreichbarkeit ist dadurch gewährleistet.

Ein weiterer Punkt in der Vorbereitungsphase ist die Überlegung der Häufigkeit der Befragung. Bei Panel-Untersuchungen werden immer diesselben Personen in regelmäßigen Abständen über denselben Tatbestand um Auskunft gebeten.

Dabei auftretende Probleme, wie die Aufrechterhaltung des Panels über den gesamten Untersuchungszeitraum, entfallen bei Einmalbefragungen. Soll eine statistische Aussage getroffen werden, ist die Größe des Stichprobenumfangs von Bedeutung. Hier wurde eine Einmalbefragung durchgeführt. Da die Gruppe, die befragt werden konnte, 17 Personen betraf, ist eine statistische Auswertung aufgrund dieser geringen Gruppengröße nicht sinnvoll.

Je nach Kommunikationskanal ist die Frager-Befragter Interaktion unterschiedlich stark standardisiert. Die am stärksten nichtstandardisierte Erhebungsmethode ist das Interview mit offener Gesprächsführung, für das lediglich ein thematischer Rahmen gegeben ist. Eine Beeinflussung ist hier sehr stark durch den Interviewer, durch dessen Äußerungen, das Erscheinungsbild, Sympathien und Antipathien gegeben. Der Einsatz dieser Form der Befragung empfiehlt sich, wenn über den zu erfragenden Themenbereich bislang wenig geforscht wurde, für Einzelfallanalysen oder für sensible Themen. Das andere Extrem ist die hohe Standardisierung. Dabei sind Wortlaut und Abfolge der Fragen vorgegeben und verbindlich für die Durchführung des Interviews.

In die Vorbereitungsphase fällt weiterhin die Festlegung des Ortes, an dem die Befragung stattfinden soll. Möglichkeiten sind die *In-Home-Befragungen*, Befragungen am Arbeitsplatz, auf der Straße oder *In-Hall-Befragungen*, bei denen die zu Befragenden an einen zentralen Ort eingeladen werden.

Es muß die Dauer der Befragung festgelegt werden. Die Länge ist dabei abhängig von Faktoren wie der zugrundeliegenden Forschungsfrage, der zur Verfügung stehenden Mittel sowie der einsetzbaren Interviewer und von den Befragten selber. Die eingesetzte Methode ist ebenfalls von Bedeutung. Bei einem persönlichen Interview kann die Dauer 60 bis 90 Minuten betragen, telefonische Interviews sollten 20 Minute nicht überschreiten, wobei die eigentliche Befragung maximal zehn Minuten dauern sollte.

Es ist ebenfalls zu überlegen, ob die Befragung mit Beobachtungsmethoden, wie z.B. die gleichzeitige Aufnahme auf Video, kombiniert werden soll. Befragungsmethoden können ebenfalls miteinander kombiniert werden.

Da die erhobenen Daten i.d.R. weiterverarbeitet werden, sind rechtliche Regelungen, wie z.B. das Bundesdatenschutzgesetz sowie die Datenschutzgesetze der einzelnen Länder zu beachten.

Konstruktion der Fragen

Nach Festlegung dieser allgemeinen Grundlagen erfolgt in der nächsten Phase die Konstruktion der Fragen. Dazu wird folgendes Vorgehen empfohlen ([vRHM94]):

- Fragensammlung erstellen
- Fragensammlung überarbeiten

- Reihenfolge der Fragen festlegen

Zunächst sollen etwa dreimal soviel Fragen gesammelt werden, wie später in der Befragung gestellt werden. Es gibt verschiedene Arten von Fragen. Neben der *direkten* Frage wird die *indirekte* Frage eingesetzt, um dem Befragten eine Antwort zu erleichtern, da er sich hinter einer anderen Person verbergen kann. Bei *offenen* Fragen muß die befragte Person selbst formulieren, da keine Antwortmöglichkeiten angeboten werden wie bei der *geschlossenen* Frage. Hier kann nur unter angebotenen Alternativen für die Antwort gewählt werden. Die befragte Person muß die geeignete Antwort in den vorgeschlagenen Alternativen wiedererkennen. Probleme entstehen, wenn die Antwortmöglichkeiten unscharf formuliert sind oder ganze Antwortkategorien fehlen. Außerdem ist zu beobachten, daß sich Probanden für Antworten aus den Alternativen entscheiden, ohne sich vorher Gedanken über die Frage gemacht zu haben. Es empfiehlt sich daher eine Kategorie „Weiß-nicht“ anzubieten, um nicht zu Antworten zu zwingen. Der klare Vorteil der geschlossenen Frage liegt in der leichten Auswertbarkeit. Abgestufte Anworten erhält man durch Skalen, die verbalisiert oder visualisiert sein können. Bei graphischen Skalen kann anhand eines Maßstabs die graphische Einordnung in einen leichter weiterzuverarbeitenden Zahlenwert übersetzt werden.

Wurden die Fragen gesammelt, schließt sich als nächster Schritt die sprachliche Überarbeitung des Fragen-Pools an. Dabei ist besonders auf folgende potentielle Probleme zu achten:

- die Fragen müssen mit größtmöglicher Verständlichkeit für die zu befragende Zielgruppe formuliert werden;
- die Tendenz vieler Personen, auf eine Frage mit „Ja“ zu antworten, muß durch eine geschickte Fragenformulierung vermindert werden;
- die Tendenz vieler Personen, die sozial erwünschten statt den wahren Antworten zu geben, muß ebenfalls durch eine geschickte Frageformulierung vermindert werden.

Um diese Fallstricke zu vermeiden, sind folgende Regeln zu beachten. Nach Sicherstellung, daß die Frage für die zugrundeliegende Fragestellung überhaupt relevant ist, muß überlegt werden, welche Schlüsse aus hypothetischen Antwortverteilungen gezogen werden können. Dabei muß überprüft werden, ob die Skalenqualität umfassend genug ist. Es müssen positive und negative Antwortmöglichkeiten vorgesehen sein. Die Frage muß grammatikalisch korrekt, sprachlich den Befragten angepaßt sein und darf nur einen Gedanken enthalten. Durch die Fragestellung darf dabei keine Antwort provoziert werden, d.h. Suggestivfragen müssen vermieden werden. Unbedingt zu vermeiden sind doppelte Verneinungen, da sie unterschiedlich von den Befragten aufgefaßt werden können.

Abschließend müssen die Fragen in eine Reihenfolge gebracht werden, wobei sie so weit wie möglich einem lockeren Gespräch ähneln sollten. Der Fragenpool sollte zunächst in die folgenden Fragegruppen unterteilt werden:

- *Einleitungsfragen* sollten leicht beantwortbar sein und das Interesse der Befragten ansprechen. Es soll der Eindruck einer Prüfungssituation abgebaut werden.
- *Sach- und Kontrollfragen* stellen den Hauptteil des Fragebogens. Bei der Reihenfolge innerhalb dieser Fragengruppe ist zu bedenken, daß vorangehende Fragen den Antwortspielraum für anschließende Fragen einschränken können, da bestimmte Vorstellungen und Denkraster aktiviert werden können. Um die Konsistenz der Auskünfte überprüfen zu können, werden leicht veränderte Kontrollfragen noch einmal an anderer Stelle aufgeführt.
- *Fragen zur Person* schließen die Befragung ab.

Liegt der Fragenkatalog in dieser ersten Fassung vor, muß er an die Befragungsart angepaßt werden.

Mögliche Befragungsarten

Es werden folgende Befragungsformen betrachtet: (i) persönliche, (ii) telefonische und (iii) schriftliche Befragung.

Bei der *persönlichen* Befragung liegt eine soziale Situation vor, da ein Interviewer mindestens einem Befragten gegenübersitzt. Die Kommunikationssignale können verbal, para- oder nonverbal sein. Die befragende Person muß das Befragungsumfeld daher so kontrollieren, daß ein möglichst guter Informationsfluß zwischen den Beteiligten gewährleistet wird.

Die Gesprächsführung hat einen großen Einfluß auf das Antwortverhalten. Der Interviewer sollte zuhören und durch geeignete verbale Signale des Interesses zu weiteren Aussagen ermuntern. Werden beim *weichen* Interview die Antworten vom Interviewer noch einmal kurz zusammengefaßt wiederholt, um dadurch die befragte Person zu weiteren Antworten zu stimulieren, ist das *harte* Interview durch eine aggressive und autoritäre Haltung des Interviewers charakterisiert. Leugnungsversuche und Abwehrmechanismen bei den Befragten sollen so überwunden werden. Beim *neutralen* Interview schließlich steht die Meßfunktion im Mittelpunkt. Der Gesprächspartner wird als gleichwertig angesehen.

Neben den verbalen Signalen wird die Befragungssituation subtil durch die para- und nonverbalen Signale beeinflußt (z.B. Erscheinungsbild des Interviewers, Alter, Körperhaltung, Klassen- oder Rassenzugehörigkeit, etc.). Insbesondere, wenn dadurch die Meinung des Interviewers deutlich wird (z.B. Heben der Augenbrauen, süffisantes Lächeln, etc.).

Auch das Befragungsumfeld beeinflußt die Befragung. Faktoren sind hier der Befragungszeitpunkt, der Befragungsort sowie die Anwesenheit Dritter.

An die Interviewer werden bei der persönlichen Befragung aus den soeben aufgeführten Gründen hohe Anforderungen gestellt. Eine Anpassung an die unterschiedlichen Gesprächspartner ist ebenso erforderlich, wie eine gute psychische Belastbarkeit, um Reaktionen des Befragten auffangen zu können. Außerdem ist eine umfassende Kompetenz über das Befragungsthema notwendig, um auch unerwartete Fragen zufriedenstellend beantworten zu können. Nach [vRHM94] ist daher eine Interviewer-Schulung unumgänglich, wodurch die persönliche Befragung als Datenerhebungsmethode aufwendig und teuer wird.

Beim *telefonischen* Interview entfallen einige Verzerrungs- und Beeinflussungsmöglichkeiten, wie z.B. Aussehen, Kleidung, Mimik, etc. Die Gesprächsteilnehmer sehen sich nicht mehr. Dafür werden paraverbale Signale, wie z.B. Sprechgeschwindigkeit, Stimmbeschaffenheit, aber auch leichte Sprachstörungen, wichtiger. Für den Interviewer ist es schwieriger einzuschätzen, ob der Interviewpartner die Frage richtig verstanden hat und ob er sich konzentriert am Interview beteiligt. Das Befragungsumfeld beim Gesprächspartner ist ebenfalls nicht einzuschätzen. Telefonanrufe kommen sehr häufig ungelegen oder werden als Scherz aufgenommen. Ankündigungsbriefe können hier zu einer verbesserten Akzeptanz führen.

Der Einsatz eines Telefoninterviews empfiehlt sich dann, wenn eine schnelle Untersuchung eines kurzen Themas durchgeführt werden soll. Besonders, wenn es um die Erhebung von Fakten geht, ist diese Form der Befragung angebracht.

Bei der *schriftlichen* Befragung schließlich existieren weder optische noch akustische Signale. Aus der Sicht des Interviewers existieren die Einschränkungen, daß weder Erläuterungen bei Unklarheiten gegeben werden können, noch gibt es eine Kontrolle über die Reihenfolge, in der die Fragen beantwortet werden.

Dem stehen bei der schriftlichen Befragung eine Reihe von Vorteilen gegenüber:

- Sie ist schnell, da in kurzer Zeit viele Personen gleichzeitig befragt werden können.
- Sie ist kostengünstig, da keine Reisekosten, Interviewerhonorare und Telefonkosten anfallen (Portokosten sind im Gegensatz dazu als gering anzusehen).
- Die befragten Personen können sich die Antworten überlegen bzw. mit anderen Personen darüber diskutieren, ehe eine Antwort erfolgt.

Ein Problem bei dieser Form der Datenerhebung ist jedoch die Quote der zurückgesandten Fragebögen (Rücklaufquote). Nach Untersuchungen schicken zwi-

schen 7 und 70 Prozent ([Fri83], zitiert nach [vRHM94]) bzw. zwischen 15 und 40 Prozent ([BEE91], zitiert nach [vRHM94]) der Befragten die Fragebögen zurück. Es kann dadurch zu Verzerrungen der ausgewählten Stichprobe führen.

Es existieren eine Reihe von Faktoren, die die Rücklaufquote beeinflussen, wie z.B. finanzielle bzw. nicht-finanzielle Anreize, Personalisierung (es wird das Gefühl vermittelt, es handelt sich um eine individuelle Befragung), Vorankündigung des Fragebogens, Frankierung des Rückkuverts, Angebot eines Feedbacks (die Ergebnisse werden dem Befragten mitgeteilt), Anspracheart des Begleitbriefes (Stil, Aufmachung, humorvoll vs. seriös, etc.), Merkmale des Fragebogens (z.B. hat die Farbe des Fragebogens einen Einfluß auf den Rücklauf ([JS83][Fox88], zitiert nach [vRHM94])), etc.

Auswertung und Präsentation

Wurden die Daten erhoben, erfolgt die Auswertung. Die aufgestellten Hypothesen werden anhand der erzielten Ergebnisse analysiert. Erfolgte die Befragung methodisch, sollte die Auswertung nur geringe Zeit in Anspruch nehmen. Sollen statistische Aussagen getroffen werden, ist eine rechnergestützte Analyse empfehlenswert. Dazu müssen die eingegangenen Daten unverfälscht in eine geeignete digitale Form übertragen werden, um von Statistikprogrammen genutzt werden zu können.

Abschließend erfolgt die Präsentation der Ergebnisse. Dazu stehen eine Vielzahl von Methoden zur Verfügung: Publikation, Vortrag, computergstützte Präsentation, etc. Die Art der Präsentation hat großen Einfluß auf die Bewertung der Befragung. Eine brillante Präsentation kann eine methodisch schlecht durchgeführte Befragung erheblich aufwerten ([vRHM94]). Andersherum kann eine schlechte Präsentation eine noch so gute methodische Arbeit entscheidend abwerten.

6.2.4 Eingesetzte Befragungsart

In dieser Arbeit sollte die Akzeptanz eines Groupwaresystems, das in einer wissenschaftlichen Arbeitsgruppe zum Einsatz kam, untersucht werden. Dadurch entfiel das Problem der Adressierung der zu befragenden Personen. Die Mitglieder der Gruppe, die mit dem prototypischen Groupwaresystem gearbeitet haben, bilden auch die Gruppe der zu befragenden Personen. Sie sind dem Interviewer namentlich bekannt.

Aufgrund der mitunter mehrjährigen Zusammenarbeit von Interviewer und befragter Person wurde keine persönliche Befragung durchgeführt, um eine zu große Beeinflussung durch die Person des Interviewers aufgrund von Sympathien oder Antipathien zu vermeiden. Da sich persönliche und telefonische Befragung

in diesem Punkt sehr ähneln, wurde auch die telefonische Befragung als Datenerhebungsmethode nicht in Betracht gezogen.[2]

Die Entscheidung fiel auf die *schriftliche Befragung*, da hier die genannten Vorteile, wie kostengünstig, parallele Befragung, keine unmittelbare Beeinflussung durch den Interviewer, zum Tragen kommen. Es wurde daher ein Fragebogen ausgearbeitet und an die betreffenden Personen verteilt.

6.3 Hypothesen

Die Umfrage hatte zum Ziel, folgende Hypothesen zu bestätigen bzw. zu widerlegen.

Mitglieder einer wissenschaftlichen Arbeitsgruppe im Bereich der molekularen Sequenzdatenanalyse werden durch das hier beschriebene und prototypisch realisierte Groupwaresystem in ihrer täglichen Arbeit unterstützt:

Hypothese 1: Das gemeinsame Gruppenziel kann besser und effizienter erreicht werden als ohne entsprechende Rechnerunterstützung.

Hypothese 2: Es verbessert sich die rechnergestützte Kommunikation zwischen den Gruppenmitgliedern.

Hypothese 3: Das wissenschaftliche Personal kann sich besser auf biologische Inhalte konzentrieren und muß sich weniger mit Fragen der Datenverarbeitung beschäftigen.

Hypothese 4: Die Möglichkeiten des Einsatzes neuer Technologien führen zu einer besseren Vereinbarkeit von Berufs- und Privatleben (Aspekt Telearbeit).

6.4 Durchgeführte Befragung

Der entworfene anonyme Fragebogen wurde an 17 Gruppenmitglieder ausgegeben, die mit den hier dargestellten CSCW-Applikationen gearbeitet haben. Insgesamt hat MIPS eine Gruppengröße von 26 Mitarbeiterinnen und Mitarbeitern. Von den 17 Fragebögen wurden 11 ausgefüllt zurückgegeben. Dies entspricht einer Quote von ca. 65%, was an der oberen Grenze der durchschnittlichen Rücklaufquote liegt (vgl. 6.2.3).

[2] Eine Person von außen zu engagieren, die die Befragung hätte durchführen können, schied aus Zeit- und Kostengründen aus.

Der entwickelte Fragebogen besteht aus drei Teilen mit insgesamt 50 Fragen: Im ersten Teil wecken leicht beantwortbare Einleitungsfragen das Interesse der Befragten (Frage 1 bis Frage 5).[3] Die Sach- und Kontrollfragen (Frage 6a bis Frage 12b) bilden den Hauptteil des Fragebogens. Dieser Hauptteil kann weiter unterteilt werden: die Fragen 6a bis 9h (insgesamt 21 Fragen) beschäftigen sich mit CSCW-Applikationen, die prototypisch entwickelt wurden; mit den Aspekten Telearbeit und Teilzeit befassen sich die Fragen 10a bis 12b (insgesamt 14 Fragen). Den Abschluß des gesamten Fragebogens bilden Fragen zur Person (Frage 13 bis Frage 22).

6.5 Ergebnisse der Befragung

Im folgenden erfolgt eine umfassende Darstellung der erzielten Ergebnisse, strukturiert nach den drei Hauptteilen des Fragebogens. Die Diskussion dieser Ergebnisse schließt sich im Unterkapitel 6.6 an.

6.5.1 Einleitungsfragen 1 bis 5

Die durchschnittliche Arbeitszeit, die vor dem Computer verbracht wird, liegt bei 95,5% (Frage 1). Der minimale Wert liegt dabei bei 85%, das Maximum 100% wurde dreimal angegeben.

Bei den Programmen, die am häufigsten angewendet werden, können zwei gleich große Gruppen extrahiert werden (Frage 2): die Gruppe von Standardsoftware (Internet Browser und Editor) mit 10 von 32 Nennungen sowie gruppenspezifische Applikationen (Anwendungen des CSCW-Systems und allgemeine Bioinformatikwerkzeuge) mit ebenfalls 10 Nennungen. Die Applikationen des Groupwaresystems (5 Nennungen) liegen dabei mit den anderen genannten Klassen von Anwendungen (mit ebenfalls jeweils 5 Nennungen) an der Spitze, d.h. sie wurden ebenso intensiv genutzt, wie etwa Netscape (Internet Browser) oder XEmacs (Editor).

Auf die Frage nach Programmen oder Diensten, die nicht zur Verfügung stehen, aber gerne benutzt werden würden, wurden individuelle Wünsche spezifischer Anwendungsaspekte genannt (Frage 3). Es konnte dabei keine allgemeine Klasse fehlender Applikationen oder Dienste abgeleitet werden.

Die Mehrheit der befragten Gruppenmitglieder verfügt über keine Präferenz hinsichtlich graphischer Bedienoberflächen oder Kommandozeilensysteme (54,5%). 36,4% sprachen sich für graphische Bedienoberflächen aus, 9,1% für Kommandozeilensysteme (Frage 4).

[3] Der Fragebogen ist im Anhang abgedruckt.

Die Frage, ob der Einsatz von Computern die Kommunikation mit den Kolleginnen und Kollegen erleichtern würde, beantworteten 63,3% der Befragten neutral bzw. negativ. Lediglich 9,1% sahen im Computereinsatz eine sehr deutliche Erleichterung (Frage 5).

6.5.2 Sach- und Kontrollfragen 6a bis 12b

Zehn der elf Gruppenmitglieder benutzen zumindest eine der realisierten CSCW-Anwendungen (Frage 6a). In Frage 6b wurden Aussagen über die Gesamtheit der realisierten CSCW-Anwendungen formuliert, die anhand einer Skala von „stimme völlig zu“ bis „stimme überhaupt nicht zu“ bewertet werden sollten. Die überwiegende Mehrheit kann sich nach Einführung des Groupwaresystems besser auf die eigentliche inhaltliche Arbeit konzentrieren (60%). Lediglich ein Gruppenmitglied sieht darin eine leichte Verschlechterung. Den Vorteil der geringeren notwendigen Detailarbeit, etwa Dateiformate betreffend, sehen 50% der Befragten. Alle Gruppenmitglieder stimmen darin überein, daß sich durch den Groupwareeinsatz die alltägliche Arbeit vereinfacht hat. Die Hälfte wählte dabei den Wert „stimme völlig zu“. Bis auf eine Ausnahme wird bestätigt, daß der Zeitaufwand zur Erledigung der Arbeit abgenommen habe. 80% glauben, daß ihnen das CSCW-System hilft, Fehler zu vermeiden. Daß die Koordination zwischen Gruppenmitgliedern unterstützt wird, beantworten 70% positiv. Kein Gruppenmitglied sagt aus, daß diese Koordination nach Einführung des Systems erschwert worden wäre.

Der Fragenkomplex 7a bis 7f befaßt sich mit der Literaturverwaltung, die von 91% der Befragten verwendet wird (Frage 7a). 87,5% der Befragten, die die Literaturverwaltung verwenden, sind mit ihr zumindest zufrieden, 42,9% davon sogar sehr (Frage 7b). Unzufrieden hat sich niemand geäußert.

Auf die Frage „Was gefällt Ihnen besonders gut an der Literaturverwaltung“ stellen die Nennungen „bedienungsfreundlich“, „schnell“ und „stabil“ mit 56,3% die klare Mehrheit der 16 abgegebenen Meinungen (Frage 7c). Weitere Angaben nennen etwa die Sicherstellung der Redundanzfreiheit, die Verfügbarkeit einer Programmierschnittstelle (API) sowie die Unterstützung verschiedener Arten von Literaturtypen. Auf die gegenteilige Frage „Was finden Sie besonders schlecht an der Literaturverwaltung“ wurden folgende Aspekte angegeben (Frage 7d): Zweimal wurde das Problem der Pflege des Datensatzes vor dem Hintergrund der großen Masse an Publikationen in diesem Wissenschaftsbereich genannt. Ein Gruppenmitglied sieht einen Nachteil darin, daß es sich um eine proprietäre Lösung handelt.

Mangelnde Funktionalität der Literaturverwaltung betrifft in erster Linie die Pflege des Datensatzes (Frage 7e). Zusätzliche Funktionen, wie etwa eine bessere Unterstützung in der automatisierten Eingabe von Zitaten oder Verknüpfungen

zu anderen Datensätzen innerhalb der Gruppe, könnten eine deutliche Verbesserung darstellen (46,2%). 30,8% der vermißten Funktionen betreffen komplexere Recherchemöglichkeiten. Lediglich ein Gruppenmitglied vermißt bei der graphischen Oberfläche eine Option (wahlweises Ausblenden des *abstracts* bei Journalzitaten). Zeitmangel ist die einzige Antwort eines Gruppenmitglieds auf die Frage, warum die Literaturverwaltung nicht verwendet werden würde (Frage 7f).

Der Fragenkomplex 8a bis 8e beschäftigt sich mit dem System *PrIAn* (vgl. 5.5.2). Lediglich vier der elf Gruppenmitglieder, die ihren Fragebogen zurückgegeben haben, benutzen dieses System (Frage 8a). Von diesen empfinden alle dieses System als zumindest komfortabel, ein Gruppenmitglied als sehr komfortabel (Frage 8b). Besonders die Beschleunigung der Eingabe neuer Proteinsequenzen wird als Hauptvorteil genannt (Frage 8c). Die Nachteile befassen sich mit der noch ausbaufähigen Funktionalität des Systems (Frage 8d), d.h. beschreiben Funktionen, die vermißt werden (Frage 8e). Lediglich ein Gruppenmitglied beklagt sich über die entwickelte graphische Bedienoberfläche. Es seien zu viele „Klicks“ notwendig, um eine neue Proteinsequenz einzufügen.

Die Fragen 9a bis 9h befassen sich mit der manuellen Wartung von Proteinsequenzobjekten. Erneut konnten vier Meinungen ausgewertet werden (Frage 9a). Die Hälfte beurteilte die Pflege als zumindest komfortabel (Frage 9b). Die beiden anderen Gruppenmitglieder empfinden sie zumindest nicht als unkomfortabel. Für diese Datenbankkomponente wurde eine pessimistische Variante zur Nebenläufigkeitskontrolle realisiert (vgl. 5.5.2). Im eingesetzten Prototyp existierte dazu keine graphische Bedienoberfläche. Auf die Frage, ob eine solche Oberfläche die Bedienung erleichtern würde, antworteten zwei Gruppenmitglieder mit „Ja“ und zwei mit „Nein“ (Frage 9c). Um möglichst geringe Anforderungen an Systeme stellen zu müssen, auf denen manuelle Bearbeitungen von Proteinsequenzobjekten durchgeführt werden können, werden Objekte zur Bearbeitung als Textdateien exportiert. Für die Bearbeitung wird von allen Gruppenmitgliedern der gleiche Editor (*XEmacs*) eingesetzt (Frage 9d). Ein Gruppenmitglied verwendet zusätzlich einen Editor des Betriebssystems *OpenVMS*. Die Hälfte ist mit ihrem Editor zufrieden, zwei Gruppenmitglieder sind weder besonders zufrieden, noch unzufrieden (Frage 9e). Als Vorteile dieses Editors werden vor allem die Möglichkeit der benutzerdefinierten Makros sowie der Möglichkeit, mehrere Dateien gleichzeitig bearbeiten zu können, genannt (Frage 9f). Die Komplexität dieses Editors und die anfänglichen Schwierigkeiten im Umgang werden als Hauptnachteile aufgeführt (Frage 9g). Genannte Funktionen, die beim eingesetzten Editor vermißt werden, werden von *XEmacs* bereitgestellt (Frage 9h). Ihre Nutzung ist jedoch nicht intuitiv.

Die verbleibenden Fragen des Hauptteils des Fragebogens (Fragenkomplexe 10 bis 12) befassen sich mit Telearbeit.

Auf die Frage nach den allgemeinen Vorteilen der Telearbeit wurden 20 Meinungen abgegeben (Frage 10a). Zehnmal wurde die größere Flexibilität genannt, z.B. die Unabhängigkeit von Arbeitszeit und persönlichem Rhythmus. Weiter wurden genannt: das Wegfallen von Fahrtwegen zur oder von der Arbeitsstätte (20%), die Erleicherung im Anstoßen oder Überwachen von automatisiert ablaufenden Prozessen im System (15%) sowie die größere Ruhe zur Arbeit (10%). Ein Gruppenmitglied nannte die Erleichterung bei der Kombination von Beruf und Familie.

Auf die Frage nach den allgemeinen Nachteilen der Telearbeit wurden 18 Argumente aufgeführt (Frage 10b). Die Hälfte benennt allgemein die Erschwerung der Teamarbeit (22%) bzw. die Erschwerung der Kommunikation mit den Kolleginnen und Kollegen (28%). Fehlende Diskussionen und damit einhergehende fehlende Anregungen betreffen 11% der abgegebenen Meinungen. Als Nachteile im häuslichen Bereich wurden aufgeführt: erhöhter Platzbedarf, mangelnde technische Voraussetzungen, nicht alle Programme seien verfügbar sowie fehlende Infrastruktur, wie z.B. Bibliothek oder Kopiergerät. Dies betrifft zusammen 22% der Aussagen. Als soziale Aspekte wurden die fehlende Trennung von Beruf und Freizeit bzw. Familie sowie das fehlende „Miteinander“ innerhalb der Arbeitsgruppe genannt (17%).

Von den elf Gruppenmitgliedern dieser Untersuchung gaben vier an, auf freiwilliger Basis Telearbeit auszuführen (Frage 11a). Keines dieser Gruppenmitglieder arbeitet unterwegs (Frage 11b). Zu Hause arbeiten drei Gruppenmitglieder zwischen einer und zwei Stunde pro Woche. Ein Gruppenmitglied wendet zehn Stunden für Telearbeit auf. In der Zeit, in der zu Hause gearbeitet wird, ist die Hälfte der befragten Personen die ganze Zeit mit dem Netzwerk der Arbeitsgruppe verbunden (Frage 11c), ein Gruppenmitglied die halbe und ein weiteres Gruppenmitglied lediglich 10% der Zeit. Außerhalb von MIPS werden fast ausschließlich Standardanwendungen, wie z.B. Netscape, MS Powerpoint, Photoshop, XEmacs, eingesetzt (Frage 11d). Zwei Gruppenmitglieder nutzen zumindest gelegentlich Anwendungen des CSCW-Systems (Literaturverwaltung, PrIAn). Die größte Unterstützung der Telearbeit durch MIPS wird ausschließlich in technischen Voraussetzungen gesehen, wie z.B. gute Internet-Anbindung, ISDN-Anschluß (Frage 11e). Als größte Hindernisse werden ebenfalls technische Aspekte aufgeführt: Anschluß belegt bzw. keine Unterstützung bei Computerproblemen (Frage 11f).

Sieben Gruppenmitglieder, die keine Telearbeit ausführen, wurden nach ihren Gründen befragt (Frage 11g). 40% der abgegebenen Meinungen betrafen das Bedürfnis des direkten Kontakts zu Kolleginnen und Kollegen. 30% sagten allgemein, daß sie kein Interesse an Telearbeit hätten. Ebenfalls 30% betrafen die fehlende technische Ausrüstung. Von diesen sieben Gruppenmitgliedern können sich 71% generell vorstellen, einen Teil ihrer Arbeit im Rahmen von Telearbeit zu leisten (Frage 11h). Von diesen wiederum würden 60% die Hälfte bzw. höch-

stens die Hälfte ihrer Arbeitszeit im Rahmen von Telearbeit erledigen (Frage 11i). Ein Gruppenmitglied nannte 20% der Arbeitszeit als obere Grenze. Als notwendige Voraussetzungen wurden von diesen Gruppenmitgliedern fast ausschließlich finanzielle bzw. technische Aspekte aufgeführt (Frage 11k): Übernahme der Telefonkosten, Bereitstellung der technischen Geräte, Betreuung der Hard- und Software (zusammen 71%). Einzelnennungen betrafen organisatorische Aspekte, wie z.B. Verbesserung des Informationsflusses (zusammen 29%).

Der abschließende Fragenkomplex zum Thema Telearbeit beschäftigt sich mit der Kommunikation zwischen Gruppenmitgliedern, die Teilzeit- bzw. Telearbeit betreiben (Fragen 12a und 12b). Hauptkommunikationsmittel, das von allen Gruppenmitgliedern genannt wurde, ist die elektronische Post (Frage 12a). Zweimal wurde das Telefon genannt. Einmal wurde die indirekte Kommunikation über gemeinsame Objekte des CSCW-Systems aufgeführt. Auf die Frage, wie die rechnergestützte Kommunikation verbessert werden könnte, wurden fünf Meinungen abgegeben, darunter drei Verbesserungsvorschläge (Frage 12b). Es wurde der Einsatz von „talk“ zur Unterstützung direkter Kommunikation angeregt. Ferner sind zentrale Dokumente wünschenswert (z.B. gruppeninterner Kalender, Planungen und Stand von Projekten) sowie verbesserte Annotationssysteme, aus denen die Arbeit der Kolleginnen und Kollegen hervorgeht. Ein Gruppenmitglied nannte Gespräche am Runden Tisch als vorteilhaft gegenüber rechnergestützter Kommunikation. Ein anderes Gruppenmitglied ist mit der existierenden rechnergestützten Kommunikation vollkommen zufrieden.

6.5.3 Fragen zur Person 13 bis 22

Den Abschluß des Fragebogens bildeten Fragen zur Person. Von den elf Gruppenmitgliedern, die einen Fragebogen abgegeben haben, sind sieben weiblichen und vier männlichen Geschlechts (Frage 13). Das Durchschnittsalter beträgt 37 Jahre mit einer Spanne von 29 bis 47 Jahren (Frage 14).

Bis auf ein Gruppenmitglied verfügen alle über einen Hochschulabschluß (Frage 15), 90% davon können zusätzlich einen Doktortitel vorweisen. Ein Gruppenmitglied besitzt einen Fachhochschulabschluß. Bis auf eine fehlende Nennung haben alle Gruppenmitglieder entweder Biologie (70%), Chemie (10%), Biochemie (10%) oder Lehramt Biologie und Chemie (10%) studiert.

Bis auf ein Gruppenmitglied, das als Muttersprache Italienisch hat, gaben alle Gruppenmitglieder Deutsch als ihre Muttersprache an (Frage 16). Alle sprechen während der Arbeit Englisch (Frage 17).

Durchschnittlich arbeiten die befragten Gruppenmitglieder bereits seit 3,9 Jahren bei MIPS (Frage 18). Die Spanne reicht hier von 5 Monaten bis 10 Jahren.

Von den untersuchten Gruppenmitgliedern stammen sieben Gruppenmitglieder aus Gruppen, die sich mit systematischen Genomsequenzierungsprojekten be-

fassen (Frage 19). Vier Gruppenmitglieder der Proteinsequenzdatenbankgruppe waren vertreten.

55% führen bei MIPS nie eine Programmiertätigkeit aus, 36% regelmäßig, 9% fast regelmäßig (Frage 20). 64% gaben schlechte oder sehr schlechte Kenntnisse im Umgang mit dem Computer an, bevor sie bei MIPS angefangen hatten (Frage 21). 27% stuften sich mittelmäßig ein, 9% hielten ihre Kenntnis für zumindest gut. Jetzt schätzen nur noch 18% ihre Kenntnisse als schlecht ein (Frage 22). 37% stufen sich nun mittelmäßig ein und 45% halten ihre Kenntnisse für zumindest gut.

6.6 Diskussion der Ergebnisse

In diesem Abschnitt werden die dargestellten Ergebnisse diskutiert sowie die unter 6.3 aufgestellten Hypothesen bewertet.

6.6.1 Gruppenzusammensetzung

Die befragte Gruppe zeichnet sich durch eine besondere Zusammensetzung aus: alle befragten Personen verfügen über Hochschul- bzw. Fachhochschulabschluß und alle Hochschulabgänger sind zusätzlich promoviert. Es handelt sich um Spezialisten, die wissenschaftlich arbeiten. Die Mitglieder befinden sich auf gleicher Hierarchiestufe und sind dem Leiter der Arbeitsgruppe unterstellt. Einzige Ausnahme ist die Gruppe der Proteinsequenzdatenbank, die über einen zusätzlichen Gruppenleiter verfügt. Die Arbeit zum Erreichen des gemeinsamen Gruppenziels wird fast ausschließlich am Rechner geleistet. Der Einsatz eines Groupwaresystems zur rechnerbasierten Kommunikationsunterstützung ist vor diesem Hintergrund gerechtfertigt. Der hohe Ausbildungsstandard aller Gruppenmitglieder erleichtert die Entwicklung eines Software-Systems, da vergleichbare Anforderungen an alle Mitglieder hinsichtlich der Bedienung gestellt werden können. Im Arbeitsalltag sprechen alle Gruppenmitglieder neben ihrer Muttersprache zusätzlich Englisch. Dadurch kann für den gruppenweiten Einsatz auf Standard-Software zurückgegriffen werden, für die z.B. nur englischsprachige Hilfetexte zur Verfügung stehen. 55% der befragten Gruppenmitglieder führen Programmiertätigkeiten aus, 45% schätzen ihre Kenntnisse im Umgang mit Rechnern als gut ein. Ein gewisses technischen Verständnis kann somit vorausgesetzt werden.

Der wissenschaftliche Hintergrund der geleisteten Arbeit betrifft die Möglichkeiten zur Kommunikationsunterstützung auf zwei Ebenen: zum einen kann das CSCW-System die Koordination der durch Rechnereinsatz geleisteten Tätigkeiten unterstützen (z.B. Konsistenzsicherung durch Nebenläufigkeitskontrolle). Inhaltliche Diskussionen auf wissenschaftlich-fachlicher Ebene, die durch synchrone

CSCW-Applikationen (z.B. Unterstützung von *face-to-face* Sitzungen) zum anderen unterstützt werden könnten, werden von den Gruppenmitgliedern nicht gewünscht. Hier wird das Gespräch am Runden Tisch bzw. an der Tafel vorgezogen. Bei den Vorschlägen zur Verbesserung der rechnergestützten Kommunikation wurden keinerlei derartige Wünsche geäußert. Es wurde vielmehr explizit aufgeführt, daß direkte Kommunikation einer rechnergestützten vorgezogen wird.

Die Einschränkung des hier entwickelten Groupwaresystems auf asynchrone Aspekte der Koordination für eine im Bereich der molekularbiologischen Sequenzdatenanalyse wissenschaftlich arbeitende Gruppe wurde durch die erzielten Umfrageergebnisse bestätigt.

6.6.2 Bewertung der Hypothesen

Bevor die Umfrage durchgeführt wurde, wurden die Ziele des Einsatzes unter 6.3 als Hypothesen formuliert. Anhand der ermittelten Umfrageergebnisse werden diese Zielvorgaben im folgenden bewertet.

Hypothese 1

In Hypothese 1 wird behauptet, das gemeinsame Gruppenziel könne durch geeignete Rechnerunterstützung besser und effizienter erreicht werden als ohne.

80% geben an, daß ihnen das Groupwaresystem helfe, Fehler zu vermeiden (vgl. Frage 6b). Zusammen mit automatisierten Mechanismen zur Konsistenzsicherung (vgl. etwa Frage 7c) wird die Qualität der erzeugten Inhalte verbessert. Eine Erhöhung der Effizienz durch die realisierten Applikationen wird ebenfalls bestätigt. Effizienzsteigerungen werden sowohl durch die entwickelten Applikationen (z.B. Literaturverwaltung (vgl. Frage 7c) und PrIAn (vgl. Frage 8c)) als auch durch allgemeine Verbesserungen des technischen Umfeldes (wie z.B. die Vermeidung von Fehlern (vgl. Frage 6b)) erreicht.
Hypothese 1 wird somit bestätigt.

Hypothese 2

In Hypothese 2 wird behauptet, die rechnergestützte Kommunikation zwischen Gruppenmitgliedern würde sich verbessern.

Alle Gruppenmitglieder stimmen darin überein, daß die Koordination zwischen den Mitgliedern unterstützt werde. Diese Koordination erfolgt als indirekte Kommunikation über Objekte des gemeinsamen Informationsraums. Rechnergestützte direkte Kommunikation erfolgt ausschließlich durch den Einsatz elektronischer Post. Ca. 20% der Gruppenmitglieder geben an, daß der Einsatz von Computern die Kommunikation mit den weiteren Mitgliedern nicht erleichtern würde.

Eine Verbesserung der rechnergestützten Kommunikation betrifft die Koordinationsunterstützung durch indirekte Kommunikation. Unterstützung direkter Kommunikation war nicht Bestandteil des Entwurfs des CSCW-Systems und kann daher durch die realisierten Prototypen auch nicht geleistet werden.

Hypothese 2 wird demnach bestätigt, betrifft jedoch nur die indirekte Kommunikation zwischen Gruppenmitgliedern.

Hypothese 3

Es wurde die Hypothese aufgestellt, wissenschaftliches Personal könne sich besser auf biologische Inhalte konzentrieren und müsse sich weniger mit Fragen der Datenverarbeitung beschäftigen.

Allgemein können sich durch den Einsatz des Groupwaresystems Gruppenmitglieder besser auf ihre inhaltliche Arbeit konzentrieren (60%) und müssen sich mit weniger datenverarbeitungstechnischer Detailarbeit befassen (50%). Alle Gruppenmitglieder stimmen darin überein, daß sich seit dem Einsatz des Groupwaresystems ihre alltägliche Arbeit vereinfacht habe. 80% geben an, daß ihnen das System helfen würde, Fehler zu vermeiden.
Hypothese 3 wird durch die erzielten Ergebnisse bestätigt.

Hypothese 4

Es wurde behauptet, daß der Einsatz neuer Technologien zu einer besseren Vereinbarkeit von Berufs- und Privatleben führen würde.

Befragt nach den allgemeinen Vorteilen der Telearbeit wurden in erster Linie die erhöhte Flexibilität und das Wegfallen von Fahrtwegen angegeben. Die Erleichterung im Anstoßen oder Überwachen von automatisiert ablaufenden Prozessen im System wurde ebenfalls mehrfach genannt. Allerdings wurde die bessere Vereinbarkeit von Beruf und Familie lediglich von einem Gruppenmitglied aufgeführt. Dies kann im Zusammenhang stehen mit der Besonderheit dieser Gruppe (z.B. geringer Anteil an Familienvätern bzw. -müttern). Entsprechende Fragen zum familiären Hintergrund wurden im Fragebogen jedoch nicht gestellt.

Generell wird als Nachteil der Telearbeit angesehen, daß die Kommunikation mit den Kolleginnen und Kollegen erschwert werden würde. Speziell fachliche Diskussionen innerhalb der Arbeitsgruppe werden als wichtiger Bestandteil der täglichen wissenschaftlichen Arbeit angesehen. Diese Art der direkten Kommunikationsunterstützung muß in einer eigenen Arbeit, die auf den Ergebnissen der hier erzielten Studie aufbauen kann, behandelt werden. Entsprechende rechnerbasierte Unterstützung ist Grundvoraussetzung für die Inanspruchnahme von Telearbeit, die sich allgemein 71% der befragten Gruppenmitglieder als Möglichkeit

vorstellen können. Es wird eine Kombination aus Telearbeit und Arbeit beim Arbeitgeber favorisiert, mit einem Anteil von ca. 40% (oder zwei Arbeitstagen pro Woche) an Telearbeit.

Notwendig für die Akzeptanz von Telearbeit sind die Schaffung technischer Voraussetzungen (z.B. häusliche Bereitstellung der notwendigen Hardware, wie etwa Rechner, ISDN-Anschluß) sowie finanzielle Aspekte (z.B. Übernahme der anfallenden Telefonkosten).

Zusammenfassend kann gesagt werden, daß bei MIPS Telearbeit nur von wenigen Mitarbeitern zu einem sehr geringen Teil der Arbeitszeit wahrgenommen wird. Bei allen Mitgliedern überwiegen die allgemein gesehenen Vorteile der Telearbeit, und die überwiegende Mehrheit kann sich Telearbeit unter gewissen technischen und finanziellen Voraussetzungen für einen kleinen Teil der Arbeitszeit vorstellen. Um eine effiziente Durchführung von Telearbeit gewährleisten zu können, müssen jedoch zusätzliche Voraussetzungen geschaffen werden. Diese betreffen die notwendige direkte Kommunikationsunterstützung zwischen Gruppenmitgliedern einer Arbeitsgruppe bzw. eines Projekts, die z.T. Telearbeit verrichten.
Hypothese 4 wird allgemein gesehen bestätigt, trifft jedoch nicht zu auf das in dieser Arbeit entwickelte CSCW-System.

6.6.3 Zusammenfassung

Die Zielvorgaben, die mit dem Einsatz des hier entwickelten und prototypisch realisierten Groupwaresystems in einer wissenschaftlichen Arbeitsgruppe gesetzt wurden, wurden durch eine Umfrage unter den beteiligten Personen überprüft. Die Auswertung der Ergebnisse ergab, daß die Aspekte, die dem Entwurf des Systems zugrunde lagen (vgl. Abschnitt 3.4), von den Gruppenmitgliedern akzeptiert wurden. Die mit dem Einsatz des prototypischen CSCW-Systems einhergehenden Verbesserungen der alltäglichen Arbeitssituation sowie der durch das System erzielten Arbeitsergebnisse werden von den befragten Mitgliedern als Zugewinn bestätigt.

Die durchgeführte Umfrage ergab jedoch auch, daß zur direkten Kommunikationsunterstützung ein Rechnereinsatz derzeit nicht erwünscht ist. Direkte Kommunikationsformen, speziell informelle fachliche Diskussionen innerhalb einer Arbeitsgruppe, werden als Grundvoraussetzung für eine erfolgreiche wissenschaftliche Tätigkeit angesehen. Diese bereits etablierten Kommunikationskulturen innerhalb wissenschaftlicher Arbeitsgruppen in der Bioinformatik erschweren Einsatz und Akzeptanz eines CSCW-Systems. Die Unterstützung in der Koordination durch indirekte Kommunikation über Objekte des gemeinsamen Informationsraums wird allerdings uneingeschränkt akzeptiert.

Kapitel 7

Zusammenfassung und Ausblick

7.1 Zusammenfassung

Ziel dieser Arbeit war die Unterstützung einer wissenschaftlichen, im Bereich der molekularbiologischen Sequenzdatenanalyse tätigen Arbeitsgruppe durch ein asynchrones CSCW-System. Für dieses, aus der Sicht von CSCW neue Anwendungsgebiet der *Bioinformatik* wurden bisher keine Untersuchungen hinsichtlich einer möglichen rechnerbasierten Unterstützung der Gruppenarbeit durchgeführt. Das Anwendungsgebiet zeichnet sich durch folgende Charakteristika aus:

- große Datenmengen müssen von einer spezialisierten, wissenschaftlichen Gruppe analysiert und verwaltet werden;
- weltweit werden Sequenzierungen kompletter Genome systematisch durchgeführt und der Datenzuwachs ist exponentiell; die Größen wissenschaftlicher Arbeitsgruppen im Bereich Bioinformatik wachsen jedoch nicht entsprechend, um die anfallenden Datenmengen mit den bisher verfügbaren Werkzeugen befriedigend analysieren zu können;
- das Wissen, das in die alltägliche Arbeit der Gruppenmitglieder einfließen soll, ist weltweit verteilt;
- Informationen aus heterogenen Datenressourcen müssen für Gruppenmitglieder transparent in lokale Informationsräume integriert werden;
- spezialisiertes Wissen individueller Gruppenmitglieder der Arbeitsgruppe muß zum Nutzen der gesamten Gruppe in Objekte des gemeinsamen lokalen Informationsraums eingebracht werden können;

- die Informatik muß für diesen Bereich der modernen Molekularbiologie ein Modell der Natur realisieren, in das neue Erkenntnisse der Wissenschaft ohne große Modifikationen integriert werden können.

Beim Entwurf eines CSCW-Systems für dieses Anwendungsgebiet müssen neben den dargestellten Charakteristika des Gebiets die Besonderheiten wissenschaftlicher Gruppen berücksichtigt werden:

- Gruppenmitglieder wissenschaftlicher Gruppen verfügen über eine hochqualifizierte Ausbildung;
- wissenschaftliches Personal hat große Freiheiten in der individuellen Arbeitsgestaltung (z.B. keine Kernzeiten, in denen eine Anwesenheit vorgeschrieben ist); es muß eine asynchrone Kommunikationsunterstützung in der Gruppe zur Erhaltung bzw. Steigerung der Effizienz ermöglicht werden.

Diese Aufzählungen verdeutlichen, daß die Bioinformatik neben dem Forschungsgebiet CSCW noch eine Reihe weiterer, bisher ungelöster Aufgaben für die Informatik als Wissenschaft bereithält. Als Beispiele seien hier die Modellierung und Simulation komplexer metabolischer und regulatorischer Netzwerke in lebenden Organismen, Annotationsmanagementsysteme zur Unterstützung in systematischen Genomsequenzierungsprojekten sowie spezialisierte Algorithmen in der Sequenzdatenanalyse genannt.

In dieser Arbeit wurde der Fokus auf die Unterstützung einer asynchron, auf einem gemeinsamen Datenraum arbeitenden Gruppe von Wissenschaftlern gelegt. Um diese Aufgabe durchführen zu können, erfolgte zunächst eine Analyse der Anforderungen. Dazu wurde die wissenschaftliche Arbeitsgruppe MIPS am Max-Planck-Institut für Biochemie, Martinsried, unter der Leitung von Herrn Prof. Dr. H.-W. Mewes untersucht. Die Anforderungen der in dieser Gruppe existierenden Untergruppen wurden zusammengestellt. Sie bildeten die Basis für den generischen Entwurf eines CSCW-Systems, das bestimmte Aspekte der Gruppenarbeit, wie etwa asynchrone, indirekte Kommunikation behandeln sollte.

Die Besonderheiten des Anwendungsgebiets, die bei der Modellierung beachtet werden mußten, wurden durch die knappe Darstellung des biologischen Hintergrunds beleuchtet. Insbesondere die Flexibilität, zukünftiges Wissen im Kontext bestehender Informationen integrieren zu können, ist eine zentrale Forderung. Der Einsatz objektorientierter Techniken ist daher die Methode der Wahl, da die zugrundeliegenden Konzepte (wie z.B. Assoziation, Aggregation, Vererbung, Polymorphismus, etc.) den notwendigen Raum für Erweiterungen eines Modells bereitstellen. Dabei wurde nicht nur bei der Applikationsentwicklung auf diese Konzepte zurückgegriffen, sondern auch auf der persistenten Datenverwaltungsebene. Es wurde die, im industriellen Umfeld bisher noch nicht etablierte Technologie der objektorientierten Datenbankmanagementsysteme eingesetzt. Die in

dieser Arbeit erzielten Erfahrungen belegen, daß kommerziell verfügbare objektorientierte Datenbankmanagementsysteme (wie z.B. *ObjectStore* von Object Design) die für den erfolgreichen Einsatz notwendigen Voraussetzungen erfüllen. Unterstützung der objektorientierten Konzepte, Skalierbarkeit, Performanz, etc. erlauben den Einsatz auch in einem Umfeld, in dem große Datenmengen verwaltet werden müssen. Besonders die einfache Integration objektorientierter Datenbanken in den objektorientierten Entwicklungsprozeß führt zu flexibleren und offeneren Softwaresystemen. Anpassungen an veränderte Rahmenbedingungen sind leichter durchführbar.

Die dem hier entwickelten CSCW-System zugrundeliegende Software-Architektur basiert auf etablierten Techniken des Software-Engineerings. Es wurden keine neuen, proprietären Lösungen für wiederkehrende Probleme im Entwurf entwickelt. Vielmehr werden die in den letzten Jahren hervorgebrachten und in der Praxis als zuverlässig eingestuften Konzepte auf die hier dargestellte Problematik übertragen (z.B. das Entwurfsmuster der Schichten, standardisierte Kommunikationstechnologien wie *Remote Procedure Call* und CORBA, etc.).

Die Kombination existierender Technologiebausteine zu individuellen Komponenten erlaubt die Realisierung neuer mächtiger und stabiler Systeme in neuen Anwendungsgebieten. Das Komponentenmodell ermöglicht die Entwicklung, Wartung und Erweiterung komplexer Systeme. Änderungen werden nur lokal an Komponenten vorgenommen, Kommunikation zwischen Komponenten geschieht via definierter Schnittstellen. Dadurch werden Netzwerke von Abhängigkeiten im System vermieden.

Basierend auf diesen grundlegenden Entwurfsentscheidungen wurde anhand eines, nach Abschluß der Analysephase aufgestellten Anforderungskatalogs ein prototypisches Groupwaresystem entworfen, implementiert und in der wissenschaftlichen Arbeitsgruppe MIPS eingesetzt. Abschließend wurde unter den beteiligten Gruppenmitgliedern eine schriftliche Umfrage durchgeführt. Ziel dieser empirischen Untersuchung war die Ermittlung der Akzeptanz des Systems, das sich im täglichen Einsatz befand.[1] Zum Zeitpunkt der Umfrage waren Teile des Systems bereits zwischen sechs Monate und drei Jahre verfügbar.

Das Ergebnis dieser Umfrage ergab eindeutig, daß sich die alltägliche Arbeit der Gruppenmitglieder durch das System deutlich verbessert hat.

7.2 Ausblick

Das in dieser Arbeit entwickelte CSCW-System behandelt nur Teilaspekte rechnergestützter Gruppenarbeit. Lediglich die asynchrone und indirekte Kommunika-

[1] Das System ist auch nach Abschluß dieser Arbeit weiterhin in der Gruppe MIPS im täglichen Einsatz.

tion über gemeinsame Objekte wird betrachtet. Die hier dargestellten und durch Einsatz sowie Akzeptanzanalyse evaluierten Konzepte zur Realisierung stellen eine Infrastruktur zur Verfügung, auf der weitere Anwendungsaspekte in der Bioinformatik entwickelt werden können.

Die durchgeführte Akzeptanzanalyse ergab, daß in der untersuchten Gruppe kein Bedürfnis nach Unterstützung in der synchronen Gruppenkommunikation besteht. Dennoch bleibt zu untersuchen, inwieweit dies im Umfeld der wissenschaftlichen Bioinformatik möglich wäre. Ein CSCW-System muß sich dabei immer der Kritik seiner AnwenderInnen stellen.

In der Modellierung der Anwendungsentitäten muß das prototypische CSCW-System dahingehend erweitert werden, kontextabhängige Informationen nicht nur verwalten und visualisieren (z.B. Abfolge genetischer Elemente auf einem Chromosom), sondern auch als Diskussionsbeitrag in den gemeinsamen Informationsraum einbringen zu können (z.B. verschiedene Genmodelle als Resultate unterschiedlicher Vorhersagealgorithmen). Es müssen spezialisierte Wissenschaftlerinnen und Wissenschaftler in der manuellen Begutachtung automatisiert erstellter Modelle unterstützt werden. Getroffene Entscheidungen müssen für alle Gruppenmitglieder zu einem späteren Zeitpunkt nachvollziehbar sein. Es müssen organisatorische Informationen (z.B. welches Laboratorium hat welchen genomischen Abschnitt sequenziert, wann sind welche Daten beim Koordinator eingetroffen, etc.) abgebildet und innerhalb des Projekts je nach Gruppenzugehörigkeit zugänglich gemacht werden.

Als weiteres Beispiel für eine zukünftige Erweiterung sei die Unterstützung der systematischen Genomsequenzierung genannt, die von mehreren Partnern durchgeführt wird. In diesem Fall erfolgt die asynchrone und indirekte Kommunikation nicht nur innerhalb einer Arbeitsgruppe, sondern zwischen allen beteiligten Partnern eines Projekts: den Laboratorien, den, auf ein Teilgebiet der Forschung spezialisierten Gruppen, den biologischen Koordinatoren und den Bioinformatikern. Diese Gruppen sind je nach Art des Projekts innerhalb eines Landes oder innerhalb eines Kontinents räumlich und zeitlich verteilt. Eine entsprechende Kommunikationsunterstützung ist Grundvoraussetzung für eine effiziente Projektabwicklung.

Das CSCW-System muß durch Öffnung nach außen erweitert werden. Dadurch wird eine Möglichkeit geschaffen, existierendes Wissen, das in der *community* existiert, in den lokalen Informationsraum des CSCW-Systems intergrieren zu können. Außerdem können Gruppen, die Partner innerhalb eines Projekts sind, verstärkt über Raum und Zeit verteilt sein und mit Unterstützung des Groupware-Systems effizient und kostengünstig zusammenarbeiten. Speziell der ökonomische Gesichtspunkt spielt in der durch Forschungsgelder finanzierten Grundlagenforschung eine entscheidende Rolle. Die für die Kommunikation innerhalb des CSCW-Systems notwendigen technischen Voraussetzungen sind vergleichsweise

kostengünstig und können in vielen weiteren Projekten wiederverwendet werden (z.B. Hardwarekomponenten wie digitale Netzwerke oder Softwarelizenzen, beispielsweise für DBMSe).

Die Öffnung nach außen bedeutet im Extremfall, daß alle Mitglieder einer bestimmten *community* an dem gemeinsamen Ziel arbeiten können. Diese Vorstellungen sind nicht rein theoretischer Natur. So wurde in den vergangenen Jahren das Genom der Pflanze *Arabidopsis thaliana* in einem weltweiten Projekt systematisch sequenziert und analysiert. Dies wird das erste vollständig bestimmte Genom einer Pflanze sein, das der Öffentlichkeit präsentiert werden kann. Innerhalb dieses Sequenzierungsprojekts bildeten sich internationale Untergruppen: so arbeiteten etwa bei der Sequenzierung und Analyse von Chromosom III Gruppen aus Europa, Japan und den USA zusammen. Als Haupthindernis, ein homogenes Resultat zu erhalten, wurden die verschiedenen eingesetzten Techniken in der Bioinformatik (z.B. verschiedene Vorhersageprogramme für Genmodelle) sowie unterschiedliche Datenformate mit heterogenen Semantiken genannt. Hier kann ein globales CSCW-System, das eine Integration internationaler Untergruppen innerhalb eines Groupwaresystems zu einer virtuellen homogenen Gruppe ermöglicht, eine großen Beitrag in der Effizienz der automatisierten und manuellen Sequenzdatenanalyse beisteuern. Kostenaufwendige Zusammenkünfte der Beteiligten können auf ein Mindestmaß reduziert werden, speziell in Projekten, in denen Vertreter der beteiligten Gruppen über mehrere Kontinente verteilt sind. Im globalen Wettbewerb, der in diesem Bereich der Gentechnologie zwischen akademischen und kommerziellen Gruppen existiert, wird durch die systematische Integration unter Beachtung der Asynchronität und Konsistenz der eingebrachten und verwalteten Informationen ein beträchtlicher Vorteil erreicht.

Digitale Netzwerke und die damit mögliche Kommunikation lassen Raum- und Zeitgrenzen verschwinden. Eine Öffnung des CSCW-Systems nach außen bedeutet jedoch, daß Sicherheitsaspekte verstärkt beachtet werden müssen. Nicht nur Authentifizierung muß sichergestellt sein, sondern auch der unauthorisierte Zugriff, z.B. auf noch nicht veröffentlichte Informationen, muß ausgeschlossen werden. Im kommerziellen wie im akademischen Umfeld ist dies speziell aus patentrechtlichen Gründen im Bereich der Genomforschung und ihrer potentiellen Anwendung (z.B. in der Medizin) ein wichtiger Aspekt für die Akzeptanz eines CSCW-Systems. Aus technischer Sicht muß beachtet werden, daß Netzsicherheitskomponenten (Firewalls) nicht nur an der Grenze zwischen internen Netzen und dem öffentlichen Internet existieren, sondern auch zwischen Gruppen eines Instituts oder Abteilungen eines Unternehmens auftreten können. Dies kann zu komplizierten Kommunikationsstrukturen führen, da z.B. bei entsprechend restriktiver Konfiguration einer Firewall eine CORBA-Kommunikation ohne zusätzlichen Aufwand nicht mehr möglich ist. Werden diese zusätzlichen Sicherheitsanforderungen umgesetzt, ist das hier beschriebene System im kommerziellen indu-

striellen Umfeld einsetzbar.

Anhang A

Fragebogen Groupware

Vielen Dank, daß Sie an der Untersuchung teilnehmen. Ich bitte Sie, ein paar Fragen zum Thema Groupware, Telearbeit und verschiedenen Systemen/Programmen aus diesem Bereich zu beantworten. Ihre Angaben werden selbstverständlich anonym ausgewertet und nur für wissenschaftliche Zwecke genutzt. Dieser Fragebogen berücksichtigt mehrere Programme, die nich alle MitarbeiterInnen nutzen. Für Sie sind daher möglicherweise nicht alle Fragen zutreffend. Vor entsprechenden Fragenteilen wird jeweils darauf hingewiesen („⇒ weiter mit Frage...“).

Frage 1: Wieviel Prozent Ihrer Arbeitszeit verbringen Sie, so im Durchschnitt über die ganze Woche gesehen, vor dem Computer?

..... %

Frage 2: Welche sind die drei häufigsten Programme oder Dienste, die Sie bei Ihrer Arbeit benutzen?

1. .....
2. .....
3. .....

Frage 3: Welche Programme oder Dienste, die Ihnen nicht zur Verfügung stehen, würden Sie gerne benutzen?

1. .....
2. .....
3. .....

Frage 4: Welche Möglichkeiten der Bedienung von Programmen bevorzugen Sie?

- ☐ Graphische Bedienoberflächen
- ☐ Kommandozeilensysteme
- ☐ Keine Präferenz

Frage 5: Erleichtert der Einsatz von Computern die Kommunikation mit Ihren Kolleginnen und Kollegen?

Ja, sehr								Nein, überhaupt nicht
☐	—	☐	—	☐	—	☐	—	☐

Im folgenden möchte ich Ihnen einige allgemeine Fragen zu den Systemen Literaturdatenbank *(Reference Database)*, Annotationsdatenbank *(PIR)* sowie *PrIAn* stellen.

Dazu finden Sie im folgenden eine Reihe von Aussagen. Geben Sie bitte für jede an, inwieweit Sie ihr zustimmen bzw. nicht zustimmen: Stimmen Sie einer Aussage *völlig* zu, dann kreuzen Sie bitte das Kästchen in der entsprechenden Zeile ganz links an. Stimmen Sie ihr *überhaupt nicht* zu, kreuzen Sie bitte das Kästchen ganz rechts an. Wenn eher ein mittlerer Wert zutrifft, wählen Sie ein mittleres Kästchen, um Ihre Meinung abzustufen.

Frage 6a: Benutzen Sie **mindestens eines** der folgenden Systeme: Literaturdatenbank *Reference Database*, Annotationsdatenbank *PIR* sowie *PrIAn*?

☐ Ja

☐ Nein (⇒ weiter mit Frage 7a)

Wenn Ja:

Frage 6b: Bitte geben Sie an, inwieweit Sie den folgenden Aussagen über **alle von Ihnen genutzten Systeme** zustimmen oder nicht.

Durch die Systeme…	stimme völlig zu				stimme überhaupt nicht zu
…kann ich mich besser auf meine eigentliche inhaltliche Arbeit konzentrieren	☐	☐	☐	☐	☐
…muß ich mich um mehr Detailarbeit (wie z.B. verschiedene Dateiformate) kümmern.	☐	☐	☐	☐	☐
…wird meine alltägliche Arbeit vereinfacht	☐	☐	☐	☐	☐
…brauche ich mehr Zeit, um meine Aufgaben zu erfüllen.	☐	☐	☐	☐	☐
…vermeide ich Fehler.	☐	☐	☐	☐	☐
…wird die Koordination zwischen mir und meinen Kolleginnen und Kollegen unterstützt.	☐	☐	☐	☐	☐

Frage 7a: Benutzen Sie – zumindest manchmal – die **Literaturverwaltung** *(Reference Database)*?

☐ Ja
☐ Nein (⇒ weiter mit Frage 7f)

Wenn Ja:

Frage 7b: Wie zufrieden sind Sie mit der **Literaturverwaltung**?

Sehr zufrieden				Überhaupt nicht zufrieden
☐	☐	☐	☐	☐

Frage 7c: Was gefällt Ihnen besonders gut an der **Literaturverwaltung**?

.....

.....

Frage 7d: Was finden Sie besonders schlecht an der **Literaturverwaltung**?

.....

.....

Frage 7e: Welche Funktion vermissen Sie bei der **Literaturverwaltung**?

.....

.....

Wenn Nein:

Frage 7f: Warum verwenden Sie die **Literaturverwaltung** nicht für Ihre Arbeit?

.....

.....

Frage 8a: Benutzen Sie – zumindest manchmal – das System **PrIAn**?

☐ Ja

☐ Nein (⇒ weiter mit Frage 9a)

Wenn Ja:

Frage 8b: Wie komfortabel finden Sie die Eingabe neuer Proteinsequenzen durch **PrIAn**?

Sehr komfortabel								überhaupt nicht komfortabel
☐	—	☐	—	☐	—	☐	—	☐

Frage 8c: Worin sehen Sie die Hauptvorteile von **PrIAn**?

.....

.....

Frage 8d: Worin sehen Sie die Hauptnachteile von **PrIAn**?

.....

.....

Frage 8e: Welche Funktion vermissen Sie bei **PrIAn**?

.....

.....

Frage 9a: Arbeiten Sie in der Annotationsgruppe der **Proteinsequenzdatenbank**?

- ☐ Ja
- ☐ Nein (⇒ weiter mit Frage 10a)

Wenn Ja:

Frage 9b: Wie komfortabel finden Sie die Pflege der Proteinsequenzdatenbank?

Sehr komfortabel									überhaupt nicht komfortabel
☐	—	☐	—	☐	—	☐	—	☐	

Frage 9c: Um Proteinsequenzen bearbeiten zu können, führen Sie jedesmal ein *checkout*– und ein *checkin*-Kommando aus. Würde eine graphische Oberfläche Ihnen die Bedienung des *checkout/checkin*-Mechanismus erleichtern?

- ☐ Ja
- ☐ Nein
- ☐ Weiß nicht

Frage 9d: Welchen Editor verwenden Sie für die manuelle Annotation?

- ☐ XEmacs
- ☐ .....

Frage 9e: Wie zufrieden sind Sie mit diesem Editor?

Sehr zufrieden									Überhaupt nicht zufrieden
☐	—	☐	—	☐	—	☐	—	☐	

Frage 9f: Worin sehen Sie die Vorteile Ihres Editors?

.....

.....

Frage 9g: Worin sehen Sie die Nachteile Ihres Editors?

.....

.....

Frage 9h: Welche Funktionen vermissen Sie bei Ihrem Editor?

.....

.....

Im folgenden möchte ich Ihnen ein paar Fragen zur **Telearbeit** generell und bei MIPS im besonderen stellen. Mit Telearbeit sind dabei alle Tätigkeiten gemeint, die Sie im Rahmen eines Projektes bei MIPS außerhalb des Instituts (z.B. zu Hause oder unterwegs) erledigen.

Frage 10a: Worin sehen Sie allgemein die Vorteile der Telearbeit?

.....

.....

Frage 10b: Worin sehen Sie allgemein die Nachteile der Telearbeit?

.....

.....

Frage 11a: Führen Sie im Rahmen Ihrer Tätigkeit bei MIPS Telearbeit aus?

- ☐ Ja
- ☐ Nein (⇒ weiter mit Frage 11g)

Wenn Ja:

Frage 11b: Wieviele Stunden pro Woche arbeiten Sie durchschnittlich zu Hause bzw. unterwegs?

..... Stunden zu Hause

..... Stunden unterwegs

Frage 11c: Wieviel Prozent der Zeit, die Sie außerhalb des Instituts arbeiten, sind Sie mit MIPS verbunden (online)?

..... %

Frage 11d: Welche Programme oder Dienste setzen Sie außerhalb von MIPS ein?

1. .....
2. .....
3. .....

Frage 11e: Worin sehen Sie die größte Unterstützung Ihrer Telearbeit durch MIPS?

.....

.....

Frage 11f: Worin sehen Sie das größte Hindernis Ihrer Telearbeit durch MIPS?

.....

.....

⇒ **weiter mit Frage 12a!**

Wenn Nein:

Frage 11g: Warum führen Sie keine Telearbeit aus?

.....

.....

Frage 11h: Können Sie sich generell vorstellen, einen Teil Ihrer Arbeit im Rahmen von Telearbeit zu leisten?

- ☐ Ja
- ☐ Nein (⇒ weiter mit Frage 12a)

Wenn Ja:

Frage 11i: Wieviel Prozent Ihrer Arbeitszeit würden Sie gerne zu Hause arbeiten?

..... %

Frage 11k: Welche Voraussetzungen müßten bei MIPS bestehen, damit Sie einen Teil Ihrer Arbeit im Rahmen von Telearbeit leisten könnten?

.....

.....

Frage 12a: Einige MIPS MitarbeiterInnen arbeiten Teilzeit, manche arbeiten von zu Hause. Auf welche computer-gestützte Art kommunizieren Sie mit Ihren Kolleginnen und Kollegen?

☐ Email

☐ .....

Frage 12b: Wie könnte die computer-gestützte Kommunikation verbessert werden?

.....

.....

Zum Abschluß noch ein paar Fragen zu Ihrer Person.

Frage 13: Ihr Geschlecht?

☐ weiblich

☐ männlich

Frage 14: Wie alt sind Sie?

.....Jahre

Frage 15: Welchen Schulabschluß besitzen Sie?

☐ Hauptschule

☐ Realschule

☐ Gymnasium

☐ Hochschulabschluß

Fach: .....

Promotion? ☐ Ja ☐ Nein

☐ anderer: .....

Frage 16: Ihre Muttersprache?

.....

Frage 17: Welche Sprache(n) sprechen Sie während der Arbeit?

.....

Frage 18: Wie lange arbeiten Sie schon bei MIPS?

..... Jahre ..... Monate

Frage 19: Bitte nennen Sie die letzten Projekte, an denen Sie bei MIPS gearbeitet haben.

1. .....
2. .....
3. .....

Frage 20: Wie oft führen Sie bei MIPS Programmiertätigkeiten aus?

regelmäßig nie

☐ — ☐ — ☐ — ☐ — ☐

Frage 21: Wie gut waren Ihre Kenntnisse im Umgang mit dem Computer, bevor Sie bei MIPS angefangen haben?

sehr gut sehr schlecht

☐ — ☐ — ☐ — ☐ — ☐

Frage 22: Wie schätzen Sie Ihre Computerkenntnisse jetzt ein?

sehr gut sehr schlecht

☐ — ☐ — ☐ — ☐ — ☐

Vielen Dank für Ihre Mitarbeit!

Falls Sie Interesse an den Ergebnissen der Untersuchung haben, dann geben Sie dies bitte unten zusammen mit Ihrer E-Mail-Adresse an. Sobald die Auswertung erfolgt ist, erhalten Sie eine E-Mail mit den Ergebnissen der Studie.

.....

□ Ja, ich möchte über die Ergebnisse dieser Umfrage informiert werden!

Meine E-Mail-Adresse: .....

Literaturverzeichnis

[ABL+99] ANDRADE, M., N. BROWN, CH. LEROY, S. HOERSCH, A. DE DARUVAR, CH. REICH, A. FRANCHINI, J. TAMAMES, A. VALENCIA, CH. OUZOUNIS und CH. SANDER: *Automated genome sequence analysis and annotation*. Bioinformatics, 15(5):391–412, 1999.

[AGM+90] ALTSCHUL, S., W. GISH, W. MILLER, E. MYERS und D. LIPMAN: *Basic Local Alignment Search Tool*. J. Mol. Biol., 215:403–410, 1990.

[BA99] BAIROCH, A. und R. APWEILER: *The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999*. Nucleic Acids Res., 27(1):49–54, 1999.

[Bak97] BAKER, S.: *CORBA Distributed Objects*. Addison-Wesley Longman Limited, 1997.

[BBL+99] BENSON, D., M. BOGUSKI, D. LIPMAN, J. OSTELL, B. OUELLETTE, B. RAPP und D. WHEELER: *GenBank*. Nucleic Acids Res., 27(1):12–17, 1999.

[BEE91] BEREKOVEN, L., W. ECKERT und P. ELLENRIEDER: *Marktforschung. Grundlagen und praktische Anwendung*. Gabler, Wiesbaden, 1991.

[BFS+98] BAILEY, L., S. FISCHER, J. SCHUG, J. CRABTREE, M. GIBSON und G. OVERTON: *GAIA: Framework Annotation of Genomic Sequence*. Genome Research, 8:234–250, 1998.

[BGM+99] BARKER, W., J. GARAVELLI, P. MCGARVEY, CH. MARZEC, B. ORCUTT, G. SRINIVASARAO, L.-S. YEH, R. LEDLEY, H.-W. MEWES, F. PFEIFFER, A. TSUGITA und C. WU: *The PIR-International Protein Sequence Database*. Nucleic Acids Res., 27(1):39–43, 1999.

[BKE99] BULT, C., D. KRUPKE und J. EPPIG: *Electronic access to mouse tumor data: the Mouse Tumor Biology Database (MTB) project*. Nucleic Acids Res., 27(1):99–105, 1999.

[Boo96] BOOCH, G.: *Objektorientierte Analyse und Design*. Addison-Wesley, 2. korrigierter Nachdruck, 1996.

[Bor91] BORGHOFF, U.: *Fehlertoleranz in verteilten Dateisystemen*. Informatik Spektrum, 14:15–27, 1991.

[BRDE99] BLAKE, J., J. RICHARDSON, M. DAVISSON und J. EPPIG: *The Mouse Genome Database (MGD): genetic und genomic information about the laboratory mouse*. Nucleic Acids Res., 27(1):95–98, 1999.

[BS98] BORGHOFF, U. und J. SCHLICHTER: *Rechnergestützte Gruppenarbeit*. Springer-Verlag, Zweite, vollständig überarbeitete und erweiterte Auflage, 1998.

[Bü98] BÜRGER, M.: *Unterstützung von Awareness bei der Gruppenarbeit mit gemeinsamen Arbeitsbereichen*. Doktorarbeit, Technische Universität München, 1998.

[BWO+96] BULT, C., O. WHITE, G OLSEN, L. ZHOU und *et al.*: *Complete Genome Sequence of the Methanogenic Archaeon, Methanococcus janischii*. Science, 273:1058–1073, 1996.

[CELS95] CASKEY, C., R. EISENBERG, E. LANDER und J. STRAUS: *Hugo statement on patenting of DNA*. Genome Digest, 2:6–9, 1995.

[Coh95] COHEN, F.: *Calculating the Secrets of Life*, Kapitel Folding the sheets: using computational methods to predict the structure of proteins., Seiten 236–271. National Academy Press, 1995.

[CZ95] CHAPMAN, D.B. und E.D. ZWICKY: *Building Internet Firewalls*. O'Reilly & Associates, Inc., 1995.

[Day78] DAYHOFF, M.: *Atlas of Protein Sequences and Structure*. National Biomedical Research Foundation, Silver Spring, Maryland, 1978.

[Doo90a] DOOLITTLE, R.: *Computers and DNA*, Kapitel What we have learned and will learn from sequence databases, Seiten 21–31. Addison-Wesley, 1990.

[Doo90b] DOOLITTLE, R.: *Molecular Evolution: Computer Analysis of Protein and Nucleic Acid Sequences*, Band 183 der Reihe *Methods in Enzymology*, Kapitel Searching through sequence databases, Seiten 99–110. Academic Press, 1990.

[DSC+80] DAYHOFF, M.O., R.M. SCHWARTZ, H.R. CHEN, L.T. HUNT, W.C. BARKER und B.C. ORCUTT: *Nucleic acid sequence bank.* Science, 209:1182, 1980.

[DSR+99] DUNHAM, I., N. SHIMIZU, B.A. ROE, S. CHISSOE und ET AL.: *The DNA sequence of human chromosome 22.* Nature, 402:489–496, 1999.

[EA93] ETZOLD, T. und P. ARGOS: *SRS – an indexing and retrieval tool for flat file data libraries.* Comput. Appl. Biosci., 9(1):49–57, 1993.

[FA95] FRISHMAN, D. und P. ARGOS: *Secondary structure assignment from atomic coordinates.* Proteins, 23:566–579, 1995.

[FA97] FRISHMAN, D. und P. ARGOS: *75% accuracy in protein secondary structure prediction.* Proteins, 27:329–335, 1997.

[FAW+95] FLEISCHMANN, R.D., M.D. ADAMS, O. WHITE, R.A. CLAYTON und *et al.*: *Whole-Genome Random Sequencing and Assembly of Haemophilus influenzae Rd.* Science, 269:496–512, 1995.

[FGW+95] FRASER, C.M., J.D. GOCAYNE, O. WHITE, M.D. ADAMS und *et al.*: *The Minimal Gene Complement of Mycoplasma genitalium.* Science, 270:397–403, 1995.

[FHLM98] FRISHMAN, D., K. HEUMANN, A. LESK und H.-W. MEWES: *Comprehensive, comprehensible, distributed and intelligent databases: current status.* Bioinformatics, 14(7):551–561, 1998.

[FM97a] FRISHMAN, D. und H.W. MEWES: *PEDANTic genome analysis.* Trends in Genetics, 13:415–416, 1997.

[FM97b] FRISHMAN, D. und H.W. MEWES: *Protein structural classes in five complete genomes.* nature structural biology, 4(8):626–628, 1997.

[FM99] FELLENBERG, M. und H.-W. MEWES: *Interpreting Clusters of Gene Expression Profiles in Terms of Metabolic Pathways.* In: *Proceedings of the German Conference on Bioinformatics*, Seiten 185–187, 1999.

[FMMG98] FRISHMAN, D., A. MIRONOV, H.W. MEWES und M. GELFAND: *Combining diverse evidence for gene recognition in completely sequenced bactarial genomes*. Nucleic Acids Res., 26:2941–2947, 1998.

[Fox88] FOX, R.: *Mail survey response rate: A meta-analysis of selected techniques for inducing response*. Public Opinion Quarterly, 52(4):467–491, 1988.

[Fri83] FRIEDRICHS, J.: *Methoden empirischer Sozialforschung*. Westdeutscher Verlag, 1983.

[GBB+96] GOFFEAU, A., B.G. BARRELL, H. BUSSEY, R.W. DAVIS, B. DUJON, H. FELDMANN, F. GALIBERT, J.D. HOHEISEL, C. JACQ, M. JOHNSTON, E.J. LOUIS, H.W. MEWES, Y. MURAKAMI, P. PHILIPPSEN, H. TETTELIN und S.G. OLIVER: *Life with 6000 Genes*. Science, 274:546–567, 1996.

[GHJV95] GAMMA, E., R. HELM, R. JOHNSON und J. VLISSIDES: *Design Patterns*. Addison Wesley Longman, Inc., 1995.

[GR91] GIBBS, C.A. ELLIS S.J. und G.L. REIN: *Groupware – Some Issues and Experiences*. Communications of the ACM, 34(1):38–58, 1991.

[Gru93] GRUDIN, J.: *Groupware and Cooperative Work: Problems and Prospects*. In: BAECKER, RONALD M. (Herausgeber): *Readings in Groupware and Computer-Supported Cooperative Work*, Seiten 97–105. Morgan Kaufmann Publishers, Inc., 1993.

[GS96a] GAASTERLAND, T. und CH. SENSEN: *Fully automated genome analysis that reflects user needs and preferences. A detailed introduction to the MAGPIE system architecture*. Biochimie, 78:302–310, 1996.

[GS96b] GAASTERLAND, T. und CH. SENSEN: *MAGPIE: automated genome interpretation*. Trends in Genetics, 12(2):76–78, 1996.

[Gus97] GUSFIELD, D.: *Algorithms on Strings, Trees, and Sequences*. Cambridge University Press, 1997.

[Har97] HARRIS, N.: *Genotator: A Workbench for Sequence Annotation*. Genome Research, 7:754–762, 1997.

[Har99] HARREUS, D. (Herausgeber): *Gentechnologie*. Propyläen Forum. Ullstein Buchverlage GmbH & Co. KG, 1999.

[Hen91] HENNINGER, S.: *Computer Systems Supporting Cooperative Work: A CSCW'90 Trip Report*. SIGCHIS Bulletin, 23(3), Juli 1991.

[Heu96] HEUMANN, K.: *Biologische Sequenzdatenanalyse großer Datensätze basierend auf Positionsbaumvarianten*. Doktorarbeit, Technische Universität München, 1996.

[HH92] HENIKOFF, S. und J. HENIKOFF: *Amino acid substitution matrices from protein blocks*. Proc. Natl. Acad. Sci. USA, 89:10915–10919, November 1992.

[HLLR98] HELT, G., S. LEWIS, A. LORAINE und G. RUBIN: *BioViews: Java-Based Tools for Genomic Data Viualization*. Genome Research, 8:291–305, 1998.

[J.D88] J.D., ULLMAN: *Principles of Database and Knowledge-Base Systems, Vol. I.* Computer Science Press, 1988.

[Joh88] JOHANSEN, R.: *Groupware: Computer Support for Business Teams*. The Free Press - Macmillan, New York, 1988.

[JS83] JOBBER, D. und S. SANDERSON: *The effects of a prior letter and coloured questionnaire paper on mail survey response rates*. Journal of the Market Research Society, 25(4):339–349, 1983.

[Kap95] KAPS, A.: *Konsistenzsicherung in einem verteilten objektorientierten Datenbanksystem*. Diplomarbeit, Technische Universität München, 1995.

[KG00] KANEHISA, M. und S. GOTO: *KEGG: Kyoto Encyclopedia of Genes and Genomes*. Nucleic Acids Research, 28(1):27–30, 2000.

[KHF+97] KAPS, A., K. HEUMANN, D. FRISHMAN, M. BÄHR und H.W. MEWES: *Visualization and Analysis of the Complete Yeast Genome*. In: *Lecture Notes in Computer Science 1278*, Seiten 178–188. Springer Verlag, 1997.

[KHMM96] KAPS, A., K. HEUMANN, A. MAIERL und H.W. MEWES: *Genomanalyse und WWW: Vom Klon zum Klick*. Informationstechnik und Technische Informatik, 38(5):8–15, 1996.

[KKD84] KLEIN, P., M. KANEHISA und C. DELISI: *Prediction of protein function from sequence properties: a discriminant analysis of a database*. Biochim. Biophys. Acta, 787:221–226, 1984.

[KLS92] KOLAKOWSKI, L., J. LEUNISSEN und J. SMITH: *ProSearch: fast searching of protein sequences with regular expression patterns related to protein structure and function*. Biotechniques, 13:919–921, 1992.

[Koc96] KOCH, M.: *Unterstützung kooperativer Dokumentenberbeitung in Weitverkehrsnetzen*. Doktorarbeit, Technische Universität München, 1996.

[KRP+99] KARP, P., M. RILEY, S. PALEY, A. PELLEGRINI-TOOLE und M. KRUMMENACKER: *EcoCyc: Encyclopedia of Escherichia coli genes and metabolism*. Nucleic Acids Res., 27(1):55–58, 1999.

[Lau98] LAURENT, S.: *XML: A Primer*. MIS:Press, 1998.

[LDS91] LUPAS, A., M. VAN DYKE und J. STOCK: *Predicting Coiled Coils from Protein Sequences*. Science, 252:1162–1164, 1991.

[Len96] LENGAUER, T.: *Molekulare Bioinformatik*. In: I., WEGENER (Herausgeber): *Highlights aus der Informatik*, Seiten 83–111. Springer Verlag, 1996.

[MAB+97] MEWES, H.W., K. ALBERMANN, M. BÄHR, D. FRISHMAN, A. GLEISSNER, J. HANI, K. HEUMANN, K. KLEINE, A. MAIERL, S.G. OLIVER, F. PFEIFFER und A. ZOLLNER: *Overview of the yeast genome*. Nature, 387(6632):7–65, 1997.

[Mar57] MARX, K.: *Das Kapital*. Alfred Kröner Verlag Stuttgart, 1957.

[Mar99] MARSHALL, E.: *A High-Stakes Gamble on Genome Sequencing*. Science, 284:1906–1909, 1999.

[MFG+00] MEWES, H.W., D. FRISHMAN, C. GRUBER, B. GEIER, D. HAASE, A. KAPS, K. LEMCKE, G. MANNHAUPT, F. PFEIFFER, C. SCHUELLER, S. STOCKER und B. WEIL: *MIPS: a database for genomes and protein sequences*. Nucleic Acids Res., 28(1):37–40, 2000.

[MHK+99] MEWES, H.W., K. HEUMANN, A. KAPS, K. MAYER, F. PFEIFFER, S. STOCKER und D. FRISHMAN: *MIPS: a database for genomes and protein sequences*. Nucleic Acids Res., 27(1):44–48, 1999.

[Mic99] MICHAL, G. (Herausgeber): *Biochemical Pathways*. Spektrum Akademischer Verlag, 1999.

[MM93] MANBER, U. und G. MYERS: *Suffix arrays: a new method for on-line search*. SIAM J. Comput., 22:935–948, 1993.

[MPH97] MEWES, H.W., F. PFEIFFER und K. HEUMANN: *Sequenzdatenbanken: Vom Gen zum Genom*. Biospektrum, 3(2):26–31, 1997.

[MRDV99] MEDIGUE, C., F. RECHENMANN, A. DANCHIN und A. VIARI: *Imagene: an integrated computer environment for sequence annotation and analysis*. Bioinformatics, 15(1):2–15, 1999.

[MSW+99] MAYER, K., C. SCHÜLLER, R. WAMBUTT, G. MURPHY und ET AL.: *Sequence and analysis of chromosome 4 of the plant Arabidopsis thaliana*. Nature, 402:769–777, 1999.

[MT98] MURAKAMI, K. und T. TAKAGI: *Gene recognition by combination of several gene-finding programs*. Bioinformatics, 14(8):665–675, 1998.

[Nö98] NÖHMEIER, M.: *Agenten in globalen Informationsräumen*. Doktorarbeit, Technische Universität München, 1998.

[PA94] PERSSON, B. und P. ARGOS: *Predicition of transmembrane segments in proteins utilising multiple sequence alignments*. J. Mol. Biol., 237:182–192, 1994.

[Pea90] PEARSON, W.: *Rapid and Sensitive Sequence Comparison with FASTP and FASTA*. Methods in Enzymology, 183:63–98, 1990.

[RBP+93] RUMBAUGH, J., M. BLAHA, W. PREMERLANI, F. EDDY und W. LORENSEN: *Objektorientiertes Modellieren und Entwerfen*. Carl Hanser Verlag München, 1993.

[RMS+98] REICHWALD, R., K. MÖSLEIN, H. SACHENBACHER, H. ENGLBERGER und S. OLDENBURG: *Telekooperation*. Springer-Verlag, 1998.

[Sch92a] SCHILL, A.: *Remote Procedure Call: Fortgeschrittene Konzepte und Systeme - ein Überblick (Teil 1: Grundlagen)*. Informatik Spektrum, 15:79–87, 1992.

[Sch92b] SCHILL, A.: *Remote Procedure Call: Fortgeschrittene Konzepte und Systeme - ein Überblick (Teil 2: Erweiterte RPC-Ansätze)*. Informatik Spektrum, 15:145–155, 1992.

[SGR99] SLAMA, D., J. GARBIS und P. RUSSELL: *Enterprise CORBA*. Prentice Hall PTR, 1999.

[SHE95] SCHNELL, R., P. HILL und E. ESSER: *Methoden der empirischen Sozialforschung*. R. Oldenbourg Verlag, 1995.

[SMGT99] SUGAWARA, H., S. MIYAZAKI, T. GOJOBORI und Y. TATENO: *DNA Data Bank of Japan dealing with large-scale data submission*. Nucleic Acids Res., 27(1):25–28, 1999.

[STLS99] STOESSER, G., M. TULI, R. LOPEZ und P. STERK: *The EMBL Nucleotide Sequence Database*. Nucleic Acids Res., 27(1):18–24, 1999.

[SZH+99] SCHARFE, C., P. ZACCARIA, K. HOERTNAGEL, M. JAKSCH, T. KLOPSTOCK, R. LILL, H. PROKISCH, K.-D. GERBITZ, H.W. MEWES und T. MEITINGER: *MITOP: database for mitochondria-related proteins, genes and diseases*. Nucleic Acids Res., 27(1):153–155, 1999.

[THG94] THOMPSON, J.D., D.G. HIGGINS und T.J. GIBSON: *CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice*. Nucleic Acids Res., 22:4673–4680, 1994.

[TSMB95] TEUFEL, S., CH. SAUTER, TH. MÜHLHERR und K. BAUKNECHT: *Computerunterstützung für die Gruppenarbeit*. Addison Wesley Longman, Inc., 1995.

[TWK+97] TOMB, J.-F., O. WHITE, A.R. KERLAVAGE, R.A. CLAYTON und *et al.*: *The complete genome sequence of the gastric pathogen* Helicobacter pylori. Nature, 388:539–547, 1997.

[Ukk95] UKKONEN, E.: *On-line construction of suffix-trees*. Algorithmica, 14:249–260, 1995.

[vRHM94] ROSENSTIEL, L. VON, C. HOCKEL und W. MOLT: *Handbuch der Angewandten Psychologie*. ecomed verlagsgesellschaft AG & Co.KG, 1994.

[Wad98] WADMAN, M.: *Human genome deadline cut by two years*. Nature, 395, September 1998.

[Wag95] WAGNER, M.: *Groupware und neues Management*. Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, 1995.

[Wei73] WEINER, P.: *Linear pattern matching algorithms*. In: *Proc. of the 14th IEEE Symp. on Switching and Automata Theory*, Seiten 1–11, 1973.

[WF93] WOOTTON, J. und S. FEDERHEN: *Statistics of local complexity in amino acid sequences and sequence databases*. Computer & Chemistry, 17:149–163, 1993.

[WH92] WALLACE, J.C. und S. HENIKOFF: *PATMAT: a searching and extracting program for sequence, pattern and block queries and databases*. Comput. Appl. Biosci., 8:249–254, 1992.

[WK97] WALKER, D. und E. KOONIN: *SEALS: A System for Easy Analysis of Lots of Sequences*. In: GAASTERLAND, T., P. KARP, K. KARPLUS, CH. OUZOUNIS, CH. SANDER und A. VALENCIA (Herausgeber): *Fifth International Conference on Intelligent Systems for Molecular Biology (ISMB'97)*, Seiten 333–339, June 1997.

[Zin86] ZINK, CH. (Herausgeber): *Pschyrembel Klinisches Wörterbuch*. Walter de Gruyter, 255., völlig überarbeitete und stark erweiterte Auflage Auflage, 1986.