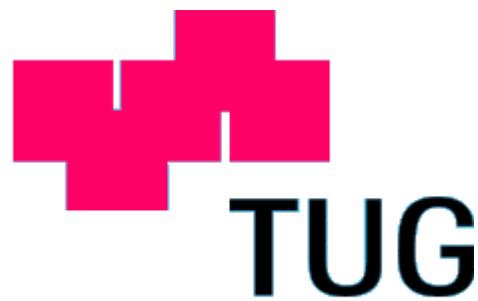


Mathias Kals

**Webinterface für YPL.db
(Yeast Protein Localization Database)**

Diplomarbeit



**Technische Universität Graz
Erzherzog-Johann-Universität**

Institut für Elektro- und Biomedizinische Technik,
Technische Universität Graz
Inffeldgasse 18, 8010 Graz
Vorstand: Univ.-Prof. Dipl.-Ing. Dr. techn. Gert Pfurtscheller

Betreuer:
Dipl.-Ing. Gerhard Thallinger,
Univ.-Prof. Dr. Sepp Kohlwein
und Dipl.-Ing. Klaus Natter

Begutachter:
Univ.-Prof. Dipl.-Ing. Dr. techn. Zlatko Trajanoski

September 2003



Institut für Molekularbiologie, Biochemie & Mikrobiologie,
Universität Graz
Schubertstrasse 1, 8010 Graz

Kurzfassung

Die YPL.db (Yeast Proteine Localization Database) (Habeler et al., Nucleic Acids Res. 2002 Jan 1;30(1):80-3) ist eine Datenbank für das Management von Daten die bei Lokalisierungsexperimenten mittels GFP (Green Fluorescent Protein) Fusionierungs-Technik und Fluoreszenz-Konfokalmikroskopie generiert werden. Im Rahmen dieser Diplomarbeit wurde ein Webinterface für die YPL.db implementiert.

Das Webinterface bietet die Möglichkeit experimentelle Bilddaten in der Datenbank zu speichern und abzufragen. Die Basis hierfür bildet das Oracle PL/SQL-TOOLKIT. Mit deren Hilfe können Datenbankabfragen bzw. Datenbankmanipulationen graphisch, mittels dynamisch generierten Webseiten, aufbereitet werden. Zur Unterstützung der Experimentauswertung wurde im Abfrageinterface eine Bildüberlagerungsfunktionalität implementiert. Mit deren Hilfe können Fluoreszenzbilder über Transmissionsbilder mit variabler Transparenz gelegt werden. Des weiteren wurde auch die Möglichkeit 3D-Bilder einzubinden implementiert. Um die Benutzung zu vereinfachen wurde das Design des Abfrageinterfaces neu gestaltet und erweitert.

Es wurde mit diesem Interface eine einfache webbasierte Möglichkeit geschaffen, Fluoreszenz-Konfokalmikroskopie-Daten zu sammeln und zu verwalten.

Schlüsselwörter: Hefe, Fluoreszenz-Konfokalmikroskopie, Protein, Lokalisierung, Oracle, PL/SQL-TOOLKIT, Bioinformatik, Bilddatenbank, 3D.

Abstract

The YPL.db (Yeast Proteine Localization Database) (Habeler et al., Nucleic Acids Res. 2002 Jan 1;30(1):80-3) is a database to manage data of localization experiments with the GFP (Green Fluorescent Protein) fusion technique and fluorescence-confocal microscopy. Within the scope of this thesis a web interface for YPL.db was implemented.

The web interface offers the opportunity to store and retrieve experimental image data, supported by the oracle PL/SQL-TOOLKIT. It dynamically generates web pages for graphical presentation of database queries or database manipulations. To support experiment utilization an image fading functionality was added. It permits superimposition of a transmission image and a fluorescence image, with variable transparency. Additionally, 3D-images can be linked to an experiment. To improve usability the query web interface was redesigned and enhanced.

With this interface an easy to handle, web-based possibility to collect and manage fluorescence-confocal microscopy data was created.

Keywords: Yeast, Fluorescence-Confocal Microscopy, Protein, Localization, Oracle, Bioinformatics, PL/SQL-TOOLKIT, Image Database, 3D.

Abbildungsverzeichnis

Abbildung 1: Prokaryotische Zelle [1]	3
Abbildung 2: Eukaryotische Zelle [6]	4
Abbildung 3: Peptidbindung [3]	6
Abbildung 4: Rückgrat einer Peptidkette [3]	6
Abbildung 5: Alpha-Helix [3]	7
Abbildung 6: Beta-Faltblatt [3]	7
Abbildung 7: Ausbildung der Tertiärstruktur [3]	8
Abbildung 8: Einbringen des GFP [16]	8
Abbildung 9: Prinzip der konfokalen Laser-Mikroskopie [9]	10
Abbildung 10: Licht Absorption und Emission	11
Abbildung 11: Auflösungsvermögen [16]	11
Abbildung 12: Prinzip der Voxel Generierung	12
Abbildung 13: VRML-Plugin für Webbrowser	13
Abbildung 14: Datenbanksystem	14
Abbildung 15: Datenbankentwurf mit ER-Modell [13]	25
Abbildung 16: Übersicht über OAS [14]	26
Abbildung 17: GUI des Oracle Form Builder	28
Abbildung 18: Gefundene Usability-Probleme [15]	33
Abbildung 19: YPL Schema	35
Abbildung 20: WYPL Schema	36
Abbildung 21: YPL.db II Login Maske	37
Abbildung 22: Benutzerverwaltung aufrufen	37
Abbildung 23: Benutzerverwaltungstabelle	38
Abbildung 24: Passwort ändern	38
Abbildung 25: YPL.db II Hauptmaske	39
Abbildung 26: Menü Bereich	39
Abbildung 27: Konstrukt-Bereich	40
Abbildung 28: Experimentdaten-Bereich	40
Abbildung 29: Bilddaten-Bereich	41
Abbildung 30: Lokalisierungsdaten-Bereich	41
Abbildung 31: Bildanzeige Bereich	42
Abbildung 32: Webexperiment-Import Übersicht	43
Abbildung 33: WWW Import Image Fenster	44

Abbildung 34: Web-Experiment-Import Fenster	44
Abbildung 35: Stammdaten-Maske	45
Abbildung 36: Konfigurations-Maske	45
Abbildung 37: YPL.db II Abfrage Seite	46
Abbildung 38: Experiment-Übertragungs-Login	47
Abbildung 39: Experiment-Übertragungs-Eingabe	47
Abbildung 40: Graphische Auswertung des Usabilitytests	51

Tabellenverzeichnis

Tabelle 1: Intrazelluläre Organellen und Kompartimente [1][3]	1
Tabelle 2: Makromoleküle [1][3]	2
Tabelle 3: Einsatz eines Datenbanksystems oder eines Dateisystems [10]	16
Tabelle 4: Struktur eines relationalen Modells	17
Tabelle 5: Übersicht über Operatoren in der Relationalalgebra [11]	18
Tabelle 6: Übersicht über Datenbankterminologie [12]	24
Tabelle 7: PL/SQL-Toolkit Pakete [14]	27
Tabelle 8: 1. Usabilitytest: Login	48
Tabelle 9: 2. Usabilitytest: Experiment übertragen	49
Tabelle 10: 3. Usabilitytest: Gen-Suche	49
Tabelle 11: 4. Usabilitytest: Lokalisierungs-Suche	49
Tabelle 12: 5. Usabilitytest: Durch Suchergebnis navigieren	50
Tabelle 13: 6. Usabilitytest: Bildergallery erstellen	50
Tabelle 14: 7. Usabilitytest: Bilder Fading	50
Tabelle 15: 8. Usabilitytest: 3D-Bild Anzeigen	50
Tabelle 16: YPI.db II Projektstruktur	60

Glossar

CVS	Concurrent Version System
DAD	Database Access Descriptor
DB	Datenbank
DBMS	Datenbank Management System
DDL	Data Definition Language
DML	Data Manipulation Language
DNA	Desoxyribonucleic Acid (auch DNS); Die DNA ist Träger der Erbinformation. Die Substanz DNA wurde 1869 von Friedrich Miescher entdeckt, ihre Bedeutung als Erbsubstanz erkannte A. T. Avery allerdings erst im Jahre 1944.
ER	Entity Relationship
GFP	Green Fluorescent Protein
GOI	Gene Of Interest
GUI	Graphical User Interface ; Graphische Benutzeroberfläche, wie sie beispielsweise von Windows, Linux (KDE) oder Solaris (CDE) verwendet wird.
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IDE	Integrated Development Environment
IMBM	Institut für Molekular-Biologie, Biochemie und Mikrobiologie an der Karl-Franzens-Universität Graz
JPEG	Joint Photographic Expert Group (auch JPG); JPEG bezeichnet eine verlustbehaftete Datenkomprimierung, die insbesondere für die Komprimierung von photorealistischem Bildmaterial mit hoher Farbtiefe geeignet ist
Konfokal	Optik mit gleichen Brennpunkten
LOV	List Of Values ; Listenfelder aus denen ein Feld ausgewählt werden kann.
OAS	Oracle Application Server
PL/SQL	Procedural Language/Structured Query Language ; PL/SQL ist eine von ORACLE entwickelte SQL-Erweiterung und dient der Entwicklung von CLIENT-SERVER -Anwendungen.
TIFF	Tagged Image File Format ; TIFF beschreibt ein weit verbreitetes Datei Format für Bilder, die unkomprimiert und komprimiert (verlustfrei) gespeichert werden können.

Tooltip	Text, der automatisch eingeblendet wird, falls der Cursor über einem Eingabefeld verweilt.
URL	Unified Resource Locator; Mit Hilfe des URL können sämtliche Informationen im Internet weltweit eindeutig adressiert werden.
VOXEL	Volumenelement (in Analogie zu Pixel) in Form eines Würfels. Grundbaustein für die Volumenmodellierung in Erweiterung des Rasteransatzes
VRML	Virtual Reality Modeling Language ist eine offene und plattformunabhängige Scriptssprache zur Darstellung von dreidimensionalen Welten.
WRL	Dateiendung für VRML-Dateien, den sogenannten Welten („ World “). VRML-Dateien liegen im Klartext vor.
WWW	World Wide Web; Das WWW ist ein Dienst im Internet, der 1989 von Tim Berners-Lee am CERN entwickelt wurde. Das WWW bietet eine einheitliche und durchdachte Bedieneroberfläche und die Möglichkeit der Einbindung multimedialer Komponenten.
YPL.db	Yeast Protein Localization database
W3C	World Wide Web Consortium; Entwickeln Spezifikationen und Guidelines für das Web.

Inhaltsverzeichnis

1	<i>Einleitung</i>	1
1.1	Biologischer Hintergrund	1
1.1.1	Die Zelle	1
1.1.2	Zellstruktur	2
1.1.3	Prokaryoten	3
1.1.4	Eukaryoten	4
1.1.5	Peptide und Proteine	5
1.1.5.1	Peptidbindung	6
1.1.5.2	Proteinstruktur	7
1.1.5.3	Proteinmarkierung mittels fluoreszierenden Proteinen	8
1.1.6	Hefe	9
1.2	Technischer Hintergrund	10
1.2.1	Konfokale-Laser-Fluoreszenz-Mikroskopie	10
1.2.1.1	Prinzip Konfokale Laser-Fluoreszenz Mikroskopie	10
1.2.1.2	Fluoreszenz	11
1.2.1.3	Auflösungsvermögen	11
1.2.2	3D-Bilder	12
1.2.2.1	Erstellung von 3D-Bildern	12
1.2.2.2	Anzeigen der 3D-Bilder	13
1.2.3	Datenbanken	14
1.2.3.1	Einführung	14
1.2.3.2	Funktionen eines Datenbanksystems	14
1.2.3.3	Einsatz von Datenbanksystemen	15
1.3	Aufgabenstellung	16
2	<i>Methoden</i>	17
2.1	Datenbanken	17
2.1.1	Das relationale Datenmodell	17
2.1.1.1	Einige Definitionen	18
2.1.1.2	Relationenalgebra	18
2.1.2	Datenintegrität	23
2.1.2.1	Primärschlüssel	23

2.1.2.2 Referentielle Integrität	23
2.1.2.3 Beziehungen	23
2.1.2.4 Normalisierung	23
2.1.2.5 Transaktionen und operationale Integrität	24
2.1.3 Entity-Relationship Modell	25
2.1.4 Data Definition Language (DDL)	25
2.1.5 Data Manipulation Language (DML)	25
2.2 OAS (Oracle Application Server)	26
2.2.1 Prinzip	26
2.2.2 PL/SQL-TOOLKIT	27
2.3 Programmierwerkzeuge	28
2.3.1 Oracle Developer Suite	28
2.3.1.1 Forms Builder	28
2.3.2 PL/SQL Developer	29
2.3.3 TOAD	29
2.3.4 WebAlchemy	30
2.3.5 Rational Rose	31
2.4 Usability	32
2.4.1 Kriterien	32
2.4.2 Anzahl der Tests	33
2.5 Motivation für verwendete Methoden	33
3 Ergebnisse	34
3.1 Datenbankstruktur	34
3.1.1 YPL Schema	35
3.1.2 WYPL Schema	36
3.2 Forms-Interface	37
3.2.1 Login-Maske	37
3.2.1.1 Benutzerverwaltung	37
3.2.1.2 Passwort setzen	38
3.2.2 Haupt-Maske	39
3.2.2.1 Menüleistenbereich	39
3.2.2.2 Konstruktbereich	40

3.2.2.3 Experimentdatenbereich	40
3.2.2.4 Bilddatenbereich	41
3.2.2.5 Lokalisierungsdatenbereich	41
3.2.2.6 Bildanzeigebereich	42
3.2.2.7 Listen-Bereich	43
3.2.3 Webexperiment Import Masken	43
3.2.4 Stammdaten-Maske	44
3.2.4.1 Konfigurations-Maske	45
3.3 Web-Interface	46
3.3.1 YPL.db II Abfrage Seite	46
3.3.2 YPL.db II Experiment Übertragungs-Seiten	46
3.3.2.1 Experiment-Übertragungs-Login	46
3.3.2.2 Experiment-Übertragungs-Eingabe	47
3.4 Usabilitytest	48
3.4.1 Auswertung des Usabilitytests	48
4 Diskussion	52
4.1 Sicherheit	52
4.2 Perspektiven	52
5 Literaturverzeichnis	54
6 INDEX	56
7 Anhang	59
7.1 YPL Installations-Anleitung	59
7.2 YPL Wartung & Änderungen	59
7.2.1 Forms	59
7.2.2 PL/SQL	59
7.3 Beschreibung (Projektstruktur, CVS)	60
7.4 YPL User Manual	60
7.5 YPL II Poster	61
7.6 Usability Test Formular	62

1 Einleitung

1.1 Biologischer Hintergrund

1.1.1 Die Zelle

Die Zellen spielen eine fundamentale Rolle bei allem uns bekannten Leben. Sie sind die kleinsten funktionellen Bestandteile von Organen und Geweben, die meist imstande sind, ihre Funktionen isoliert auszuüben. Die Zellen sind durch eine Zellmembran abgeschlossen und enthalten verschiedenste subzelluläre Strukturen, die es der Zelle ermöglichen deren Funktion auszuführen.

Zelluläre Funktionen sind in Organellen kompartimentiert. Diese können in unterschiedlichstem Ausmaß in Zellen vorkommen. Erythrozyten bilden dabei eine Ausnahme, bei ihnen sind keine intrazelluläre Strukturen nachweisbar. [1][3]

WICHTIGE INTRAZELLULÄRE ORGANELLEN UND KOMPARTIMENTE	
Kompartimente	Funktion
Zellkern	Informationsspeicherung und Verdopplung bei Zellteilung (Replication)
Mitochondrien	Energiekonservierung durch Atmungskette und oxidative Phosphorylierung (Gewinnung von ATP aus ADP und anorganischem Phosphat)
Lysosome/Vakuolen	Intrazelluläre Degradation vieler Makromoleküle
Glattes endoplasmatisches Reticulum (SER)	Lipide und Membranen werden synthetisiert
Raues endoplasmatisches Reticulum (RER)	Ribosomen werden gebunden, welche Proteine synthetisieren, die auf intrazelluläre Kompartimente verteilt oder aus der Zelle sezerniert werden müssen
Golgi-Apparat	Posttranskriptionale Modifikation der im RER synthetisierten Proteine und deren Verteilung. Membran- und Vesikeltransport bei Sekretionsprozessen

Tabelle 1: Intrazelluläre Organellen und Kompartimente [1][3]

Subzelluläre Organellen setzen sich wiederum aus einer großen Zahl unterschiedlicher Makromoleküle zusammen.

MAKROMOLEKÜLE	
Makromolekül	Beschreibung
Proteine	Polymere aus 20 proteinogenen Aminosäuren, die in für jedes Protein spezifischer Sequenz miteinander verknüpft sind.
Nukleinsäure	Für die Synthese von Nukleinsäuren werden vier Nukleotide benötigt. Sie bestehen aus einer Base, Desoxyribose bzw. Ribose und Phosphat.
Kohlenhydrate	Kommen als Mono-, Di-, Oligo- und Polysaccharide vor. Sie können unter Energiegewinn abgebaut oder zur Biosynthese von Nicht-Kohlenhydratstrukturen verwendet werden.
Lipide	Ester von Fettsäuren mit Glycerin bzw. Sphingosin als Alkohol, Sterole.

Tabelle 2: Makromoleküle [1][3]

1.1.2 Zellstruktur

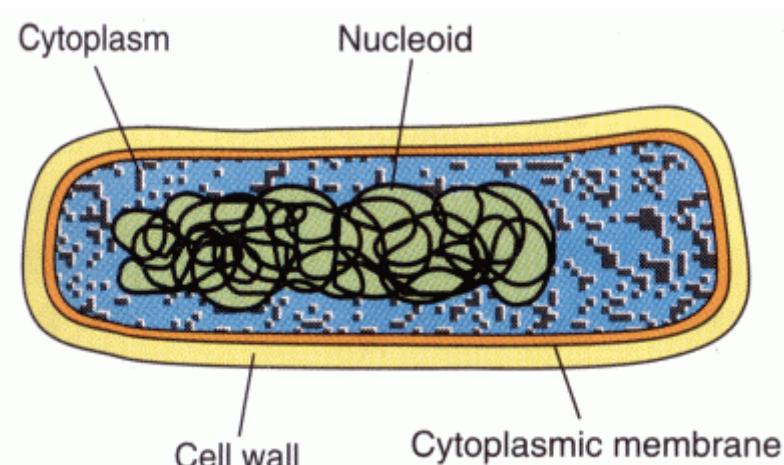
Alle Zellen besitzen eine Barriere, die Zytoplasma-Membran, die das Zytoplasma von der Umgebung abgrenzt. Die wichtigste Struktur innerhalb einer Zelle ist der Nukleus oder Nukleoid, in dem die genetische Information in Form von DNA gespeichert ist. Zellen können abhängig von der Organisation der DNA und anderen Eigenschaften in die zwei großen Gruppen von Prokaryoten (siehe Punkt 1.1.3) und Eukaryoten (siehe Punkt 1.1.4) eingeteilt werden.

Viren gelten nicht als Zellen, da sie sich alleine nicht replizieren können.

1.1.3 Prokaryoten

Bei den Prokaryoten (von *pro* d.h. vor und *karyon*, der Kern) handelt es sich um einzellige Lebewesen. Sie besitzen keinen Zellkern. Die DNA liegt als Nukleoid gepackt im Zellplasma vor. Prokaryoten besitzen eine Plasmamembran und eine Zellwand (siehe Abbildung 1: Prokaryotische Zelle). Sie sind wesentlich kleiner als Eukaryoten. Zu den Prokaryoten zählen die Bakterien und die Cyanobakterien (früher Blaualgen genannt).

Bakterien haben eine große Bedeutung in der Molekularbiologie und Medizin. Zum einen sind sie Forschungsobjekt für das Studium von Lebensvorgängen auf molekularer Ebene und zum andern Symbionten und Krankheitserreger beim Menschen.

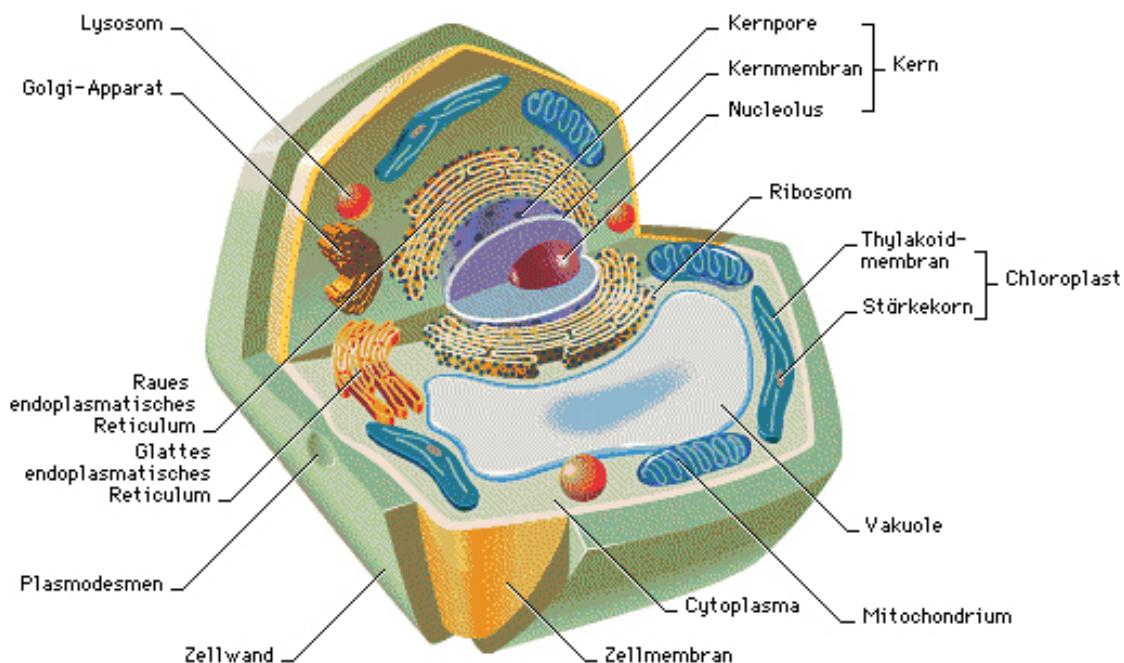


Zellmembran (Cytoplasmic membrane):	Sie umhüllt das Zytoplasma der Zelle, es können nur ungeladene Stoffe und Stoffe mit einem Molekulargewicht von weniger als 1000 durch diffundieren. Andere Stoffe werden durch spezifische Membranproteine transportiert, dies kann auch gegen einen Konzentrationsgradienten unter Aufwendung von Energie (ATP) geschehen (aktiver Transport). Die Zellmembran besteht etwa zu gleichen Teilen aus Lipiden und Proteinen. [2]
Zellwand (Cell wall):	Sie ist starr und verleiht der Zelle mechanische Stabilität. Ihre chemische Zusammensetzung variiert unter den Prokaryoten stark. Zellwandbestandteile von Bakterien bewirken Antikörperbildung in einem infizierten Organismus.
Nukleoid (Nucleoid):	Bakterienzellen besitzen keinen Zellkern, deshalb liegt das Chromosom, ein ringförmiges DNA-Molekül frei im Zytoplasma.
Zellplasma (Cytoplasm):	Das gesamte Volumen der Zelle innerhalb der Zellmembran nennt man Zellplasma. Das Zellplasma ist der Ort vieler wichtiger Lebensfunktionen der Zelle; u.a. finden hier die ersten Reaktionen des Nährstoffabbaus statt, und auch viele der großen Moleküle, aus denen die Zelle besteht, werden hier zusammengesetzt.

Abbildung 1: Prokaryotische Zelle [1]

1.1.4 Eukaryoten

Eukaryoten (gebildet aus den griechischen Wörtern *eu*, d.h. gut, wahrhaftig, und *karyon*, der Kern) besitzen einen komplexeren Aufbau als Prokaryoten (siehe Abbildung 2: Eukaryotische Zelle [6]). Sie haben einen Zellkern, in dem sich die DNA befindet, und weitere funktionelle Bereiche, die durch Membranen von Zellplasma abgegrenzt sind. Man nennt diese Abgrenzung in funktionelle Einheiten auch Kompartimentierung. Die funktionellen Einheiten sind die sogenannten Zellorganellen. In diesen Zellorganellen finden die Stoffwechselreaktionen statt. Eukaryotische Zellen sind im allgemeinen größer als prokaryotische Zellen. Sie können als Einzeller vorkommen (Protozoen, einzellige Pilze wie Hefe), meist sind sie aber in Vielzellverbänden zusammengeschlossen. Zu den Vielzellen gehören Pilze, Pflanzen und Tiere.



Zellwand:	Die Zellwand gibt der Zelle ihre Druckfestigkeit und Form. Sie stellt einen mechanischen Schutz vor äußeren Einflüssen dar. Die Zellwand ist nicht undurchlässig. Bestimmte Stoffe können durch sie in die Zelle eindringen.
Zellmembran:	Zellmembranen sind Schranken, die für bestimmte Stoffe durchlässig sind und für andere nicht. Sie haben auch die Aufgabe von Pumpen, die Stoffe in die Zelle oder aus der Zelle bringen.
Zellplasma:	Das gesamte Volumen der Zelle außerhalb des Zellkernes nennt man Zellplasma. Das Zellplasma ist der Ort vieler wichtiger Lebensfunktionen der Zelle; u.a. finden hier die ersten Reaktionen des Nährstoffabbaus statt, und auch viele der großen Moleküle, aus denen die Zelle besteht, werden hier zusammengesetzt.

Abbildung 2: Eukaryotische Zelle [6]

Mitochondrien:	 <p>Jedes Mitochondrium ist ein meist wurstförmiges, mehrere Mikrometer langes Gebilde, das von zwei unterschiedlichen Membranen umschlossen ist. Die innere dieser beiden Hälften ist stark gefaltet.</p> <p>Mitochondrien dienen der Energieerzeugung. Jede Zelle braucht Energie für Wachstum und Fortpflanzung, und diese Energie stammt zum größten Teil aus den Mitochondrien. Da bei diesem Vorgang Sauerstoff verbraucht und Kohlendioxid produziert wird, bezeichnet man ihn auch als Zellatmung. Ohne Mitochondrien könnten Tiere und Pilze den Sauerstoff nicht verwerten. Mitochondrien gibt es in Tier und Pflanzenzellen.</p> <p>Man nimmt an, dass Mitochondrien ursprünglich Bakterien waren, die sich im Verlauf der Entwicklungsgeschichte an das Leben innerhalb der Zelle angepasst haben (Besitzen als einzige Organellenart, neben den Chloroplasten bei Pflanzenzellen, eigene DNA).</p>
Golgi-Apparat:	<p>Der Golgi-Apparat nimmt die im endoplasmatischen Reticulum (ER) gebildeten Moleküle auf, verarbeitet sie weiter und dirigiert sie dann an verschiedene Stellen in der Zelle.</p>
Endoplasmatisches Reticulum (ER):	<p>Das Endoplasmatische Reticulum (ER) ist ein in pflanzlichen und tierischen Zellen ausgebildetes Röhren- und Zisternensystem, das weite Teile des Zytosplasmas durchzieht und die Kernhülle bildet.</p> <p>Man unterscheidet glattes und raues endoplasmatisches Reticulum, dessen Membranen mit Ribosomen besetzt sind. Beide Arten dienen dem Stofftransport. Im rauen endoplasmatisches Reticulum werden Eiweiße hergestellt.</p>

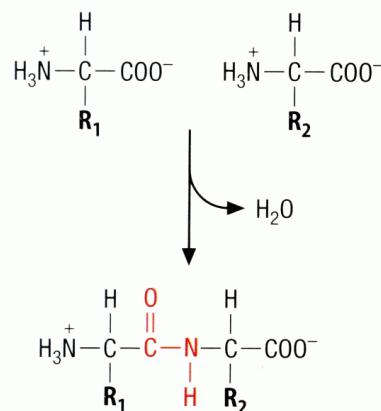
Abbildung 2 Fortsetzung: Eukaryotische Zelle [6]

1.1.5 Peptide und Proteine

Peptide und Proteine sind kettenförmige Moleküle, die aus 20 proteinogenen Aminosäuren in jeweils spezifischer Sequenz miteinander verknüpft sind. Moleküle mit 100 oder weniger Aminosäuren werden als Peptide bezeichnet, die längeren als Proteine. Diese Einteilung ist willkürlich und hat wenig Beziehung zur Funktion oder Struktur des Moleküls. Peptide und Proteine bestehen aus unverzweigten Aminosäure-Ketten, die durch Peptidbindung verknüpft sind.

1.1.5.1 Peptidbindung

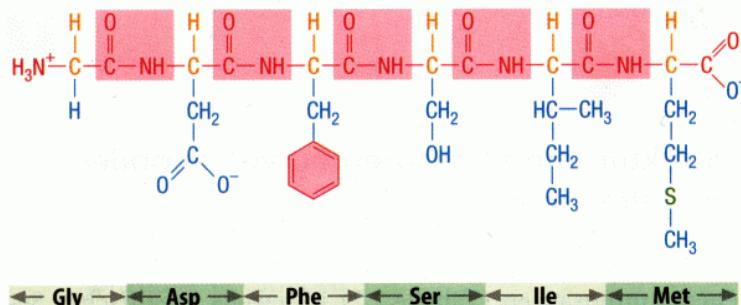
Die Peptidbindung entsteht durch Wasserabspaltung zwischen der Carboxylgruppe der einen und der Aminogruppe der nächstfolgenden Aminosäure.



Durch Wasserabspaltung zwischen der Carboxylgruppe und einer Aminosäure und der Aminogruppe der folgenden Aminosäure entsteht formal eine Peptidbindung

Abbildung 3: Peptidbindung [3]

Dadurch gehen die freien α -Amino- und α -Carboxylgruppen der Aminosäuren verloren und kommen daher nur mehr an den Enden der Peptid- bzw. Proteinkette vor.



Die einzelnen Peptidbindungen, die das Rückgrat (Hauptkette) eines Peptids bilden, sind rot gerastert, die α -C-Atome gelb und die Aminoseitenketten blau hervorgehoben

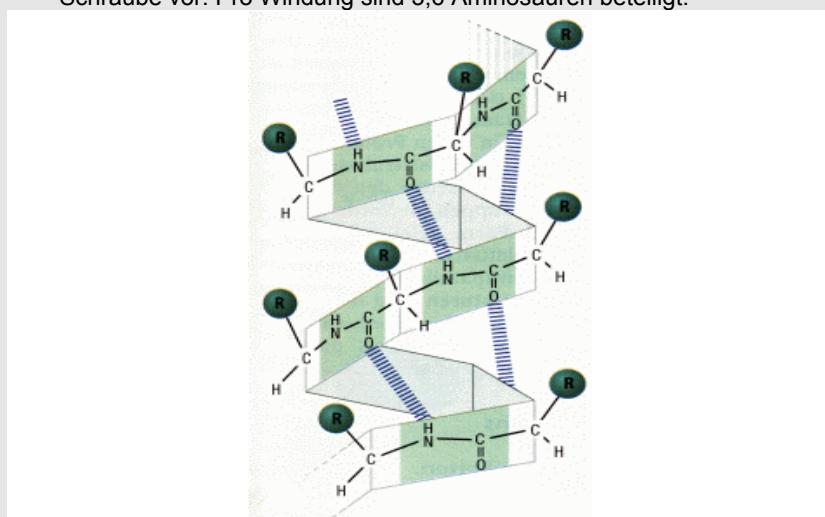
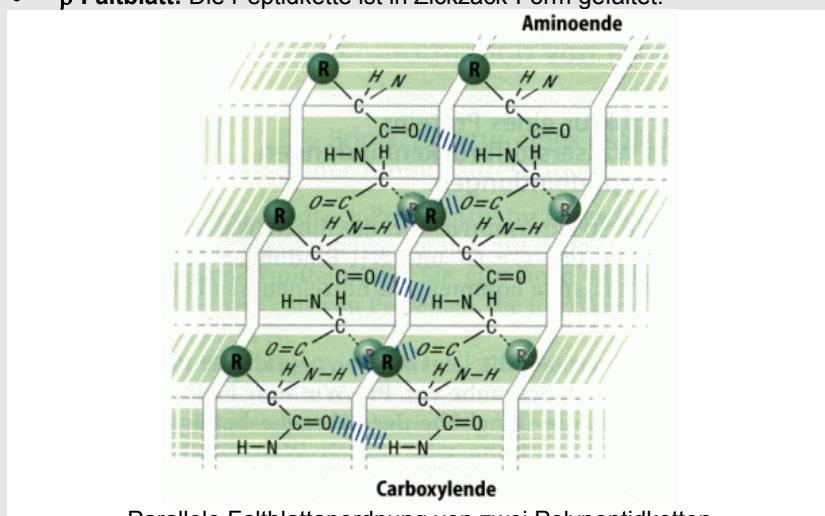
Abbildung 4: Rückgrat einer Peptidkette [3]

Alle Peptidketten besitzen folglich ein Rückgrat, welches bei allen Pepiden und Proteinen identisch ist. Die Individualität wird durch die Seitenketten, die mit dem jeweiligen α -Atom verknüpft sind, hervorgerufen. Diese Seitenketten werden aus den 20 proteinogenen Aminosäuren gebildet und sind für jedes Peptid bzw. Protein in spezifischer Sequenz verknüpft.

Geht man von einem kleinen Protein mit 100 Aminosäuren aus, so könnten mit den 20 proteinogenen Aminosäuren rein rechnerisch 20^{100} verschiedene Aminosäurensequenzen realisiert werden. In der Natur kommt jedoch nur ein Bruchteil davon vor. Man schätzt den Umfang des menschlichen Genoms auf 30000 Gene, die Proteine codieren. Durch alternative Splicing Varianten und posttranskriptionale Modifikationen ergeben sich daraus zwischen 150.000 und 300.000 verschiedene Proteine. [3]

1.1.5.2 Proteinstruktur

Proteine spielen eine Schlüsselrolle bei nahezu allen Zellfunktionen. Zwei Arten von Proteinen sind in der Zelle vertreten: Katalytische Proteine (Enzyme) und Struktur-Proteine. Enzyme wirken als Katalysator in den verschiedensten chemischen Reaktionen, die in einer Zelle ablaufen. Struktur-Proteine sind verantwortlich für die Bildung und Aufrechterhaltung räumlicher Strukturen. Sie sind integraler Bestandteil der Struktur der Zelle, der Membranen und in den Zytoskelettkomplexen [1]. Die Funktionen und Eigenschaften einzelner Proteine herauszufinden ist einer der Schwerpunkte biochemischer Forschung.

Primäre Protein-Struktur:	Entspricht der Aminosäuresequenz des Proteins. Sie gibt allerdings nur wenig Anhaltspunkte über dessen Raumstruktur, die unerlässlich für das Verständnis seiner Funktion ist.
Sekundäre Protein-Struktur:	<p>Viele Proteine enthalten Strukturmotive, die durch Wasserstoffbrücken-Bildung zwischen den C=O. und HN-Bindungen des Rückgrats von Peptidketten hervorgerufen werden.</p> <ul style="list-style-type: none"> • α-Helix: Die Polypeptidkette liegt in Form einer rechtsgewundenen Schraube vor. Pro Windung sind 3,6 Aminosäuren beteiligt.  <p>Die stabilisierenden Wasserstoff-Brücken sind blau gestrichelt</p> <p>Abbildung 5: Alpha-Helix [3]</p> <ul style="list-style-type: none"> • β-Faltblatt: Die Peptidkette ist in Zickzack-Form gefaltet.  <p>Parallele Faltblattanordnung von zwei Polypeptidketten</p> <p>Abbildung 6: Beta-Faltblatt [3]</p>

Tertiär Protein-Struktur:

Beruht auf Wechselwirkungen zwischen den Aminosäureseitenketten. Sie beschreibt die Ausbildung der stabilen Raumstruktur monomerer Proteine, die aus Kombinationen von α -Helix und β -Faltblättern sowie den dazwischen liegenden Schleifen und deren Strukturelementen bestehen. Die Struktur vieler Proteine kann in Domänen unterteilt werden. Diese besitzen definierte Funktionen.

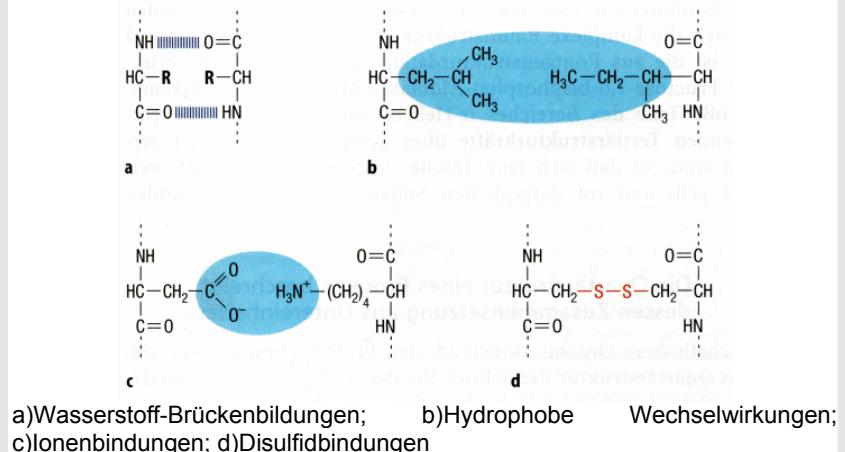


Abbildung 7: Ausbildung der Tertiärstruktur [3]

Quartär Protein-Struktur:

Beschreibt die Zusammensetzung aus Untereinheiten des Proteins.

1.1.5.3 Proteinmarkierung mittels fluoreszierenden Proteinen

Um Proteine in lebenden Zellen zu detektieren, bedient man sich der Vorteile des Green Fluorescent Protein (GFP), ein natürlich fluoreszierendes Protein, das ursprünglich aus der Qualle *Aequorea victoria* stammt.

Das Gen des GFP kann an ein anderes Gen fusioniert werden. Zellen, in die diese rekombinante DNA eingeschleust wurde, synthetisieren das entsprechende Fusionsprotein, wobei der Nachweis dieses Proteins in der Zelle anhand der grünen Fluoreszenz möglich ist. Mittels entsprechend ausgerüsteten Mikroskopen ist es möglich, die Bewegung von GFP-markierten Proteinen in der Zelle per Videoaufnahme zu dokumentieren.

Die GFP Technik wird auch angewendet, wenn keine entsprechenden spezifischen Antikörper vorhanden sind, die einen Nachweis des Proteins mittels indirekter Immunfluoreszenz ermöglichen. Ein Nachweis des GFP ist auch nach Fixierung der Zellen möglich.

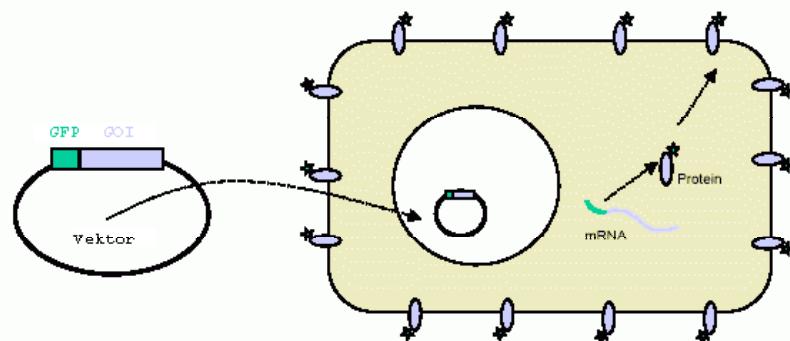


Abbildung 8: Einbringen des GFP [16]

Das Gen von Interesse (GOI) wird in den entsprechenden GFP-Expressionsvektor kloniert und zwar so, dass das GFP-Gen (hier grün) und das GOI (hier violett) im richtigen Leserahmen miteinander verbunden werden (Fusionsgen). Nachdem in vivo das rekombinante Konstrukt, mittels LiAc (Lithium Acetat bei 42° Celsius bzw. bei 315,15 Kelvin), Elektroporation oder einer anderen Methode, in die Zelle eingebracht wurde (Transfektion), kann das Fusionsgen von der Zelle exprimiert und das entsprechende Fusionsprotein (hier ein Membranprotein) gebildet werden.

1.1.6 Hefe

Hefe (*Saccharomyces cerevisiae*) spielt eine wichtige Rolle als Modellorganismus in der zellbiologischen Forschung. Zur Zeit sind die Genom-Sequenzen von zahlreichen Modellorganismen wie Bäckerhefe, Fadenwurm, Fruchtfliege, Ackerschmalwand und Maus öffentlich zugänglich. Die Vorteile von Modellorganismen liegen darin, dass die Funktion und Organisation von Proteinen leichter zu ermitteln sind. Überdies sind auch die Auswirkungen von Veränderungen bzw. Störungen auf die Nachkommen schneller und leichter zu untersuchen. Des Weiteren bestehen bei vielen menschlichen Krankheitsgenen und Stoffwechselfunktionen Homologien in der Hefe. Infolgedessen ist die Hefe hervorragend als Modellorganismus für zellbiologische Forschung geeignet.

1.2 Technischer Hintergrund

In diesem Kapitel werden die technischen Hintergründe, die dieser Diplomarbeit zu Grunde liegen, beschrieben.

1.2.1 Konfokale-Laser-Fluoreszenz-Mikroskopie

Die Konfokale-Laser-Fluoreszenz-Mikroskopie stellt eine Möglichkeit dar, die Lokalisierung von Proteinen mittels GFP (siehe Punkt 1.1.5.3) zu ermitteln. Durch diese Technik können auch dynamische Vorgänge in der Zelle beobachtet werden.

1.2.1.1 Prinzip Konfokale Laser-Fluoreszenz Mikroskopie

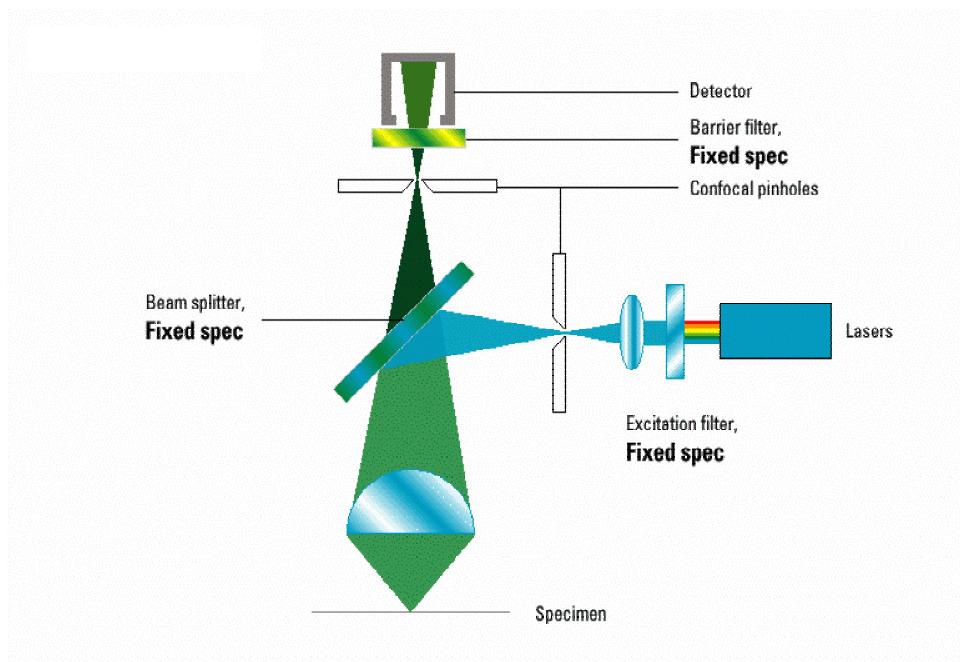


Abbildung 9: Prinzip der konfokalen Laser-Mikroskopie [9]

Da Fluoreszenzen sehr lichtschwach sind und durch das wesentlich intensivere Beleuchtungslight (Anregungslicht) des Mikroskops überstrahlt werden, ist bei der konventionellen Mikroskopie nichts von der Fluoreszenz zu sehen. Durch die Stokes-Verschiebung ist es jedoch möglich, durch Anwendung von Filtern das Anregungslicht und das Fluoreszenzlicht zu trennen.

Beim konfokalen Mikroskop wird die Überlagerung des Fluoreszenzlichts aus der Fokusebene des Objektives mit Fluoreszenzlicht, das aus Bereichen unter und oberhalb der Brennebene stammt, umgangen, indem nur Moleküle einer bestimmten Ebene visualisiert werden. Dadurch entstehen wesentlich schärfere Schnittbilder als bei konventionellen Mikroskopen. Das Auflösungsvermögen kann auf ca. 150nm erhöht werden. Das optische System fokussiert den Strahl dabei auf einen bestimmten Punkt in einer spezifischen Tiefe des Objektes. Dazu wird Laserlicht zunächst durch eine Lochblende geführt. Auch das emittierte Licht muss durch eine Lochblende, bevor es von einem Detektor (Photomultiplier), registriert wird. Beide Lochblenden, sowie ein Punkt in der Brennebene der Linse, sind konfokal. Daher

gelangt nur Licht aus der Brennebene zum Detektor, während Licht aus anderen Ebenen ausgeblendet wird. Um ein komplettes Bild zu erhalten, muss der Lichtstrahl mit Hilfe einer Scanningeinheit Punkt für Punkt über das Objekt bewegt werden. Aus den dabei aufgenommenen Einzelpunkten wird dann im Rechner ein Bild rekonstruiert.

1.2.1.2 Fluoreszenz

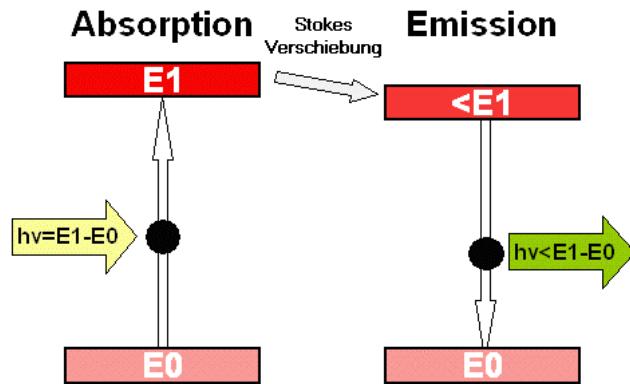


Abbildung 10: Licht Absorption und Emission

In Atomen können bestimmte Elektronen durch Absorption von Licht (Photonen) mit bestimmter Wellenlänge (Anregungslicht) in ein höheres Energieniveau gehoben werden. Diese Elektronen können sich jedoch nicht auf diesem Niveau halten und fallen praktisch im selben Augenblick in ihr ursprüngliches Energieniveau zurück. Dabei wird fast die gesamte aufgenommene Energie als Fluoreszenzlicht-Emission frei. Das Emissionslicht ist langwelliger (energiärmer), als das absorbierte Licht und besitzt daher eine andere Farbe. Diese Differenz in der Wellenlänge zwischen Absorptions- und Emissionslicht wird als Stokes-Verschiebung bezeichnet. [8]

1.2.1.3 Auflösungsvermögen

Hefezellen besitzen etwa einen Durchmesser von 5 bis $10\mu\text{m}$ (Säugerzellen $10\text{-}100\mu\text{m}$). Deshalb sind lichtmikroskopische Analysen der subzellulären Strukturen in lebenden Hefezellen experimentell aufwendig. Dennoch sind fluoreszenzmikroskopische Methoden für Hefe gut geeignet. [7]

Auftretende Probleme:

- Die maximale Auflösung bei der Wellenlänge des von GFP emittierten Lichtes entspricht etwa $0,25\text{ }\mu\text{m}$.
- Die Fluoreszenzemission der Bereiche der Zelle, die nicht in der fokalen Fläche liegen, produzieren Unschärfen.
- Interferenzen durch Beugung entstehen durch die starke Hefezellwand.
- Die Bewegung der gelösten Zellen und die interne Dynamik der Strukturen in der Zelle ergeben weitere Beschränkungen der Auflösung.

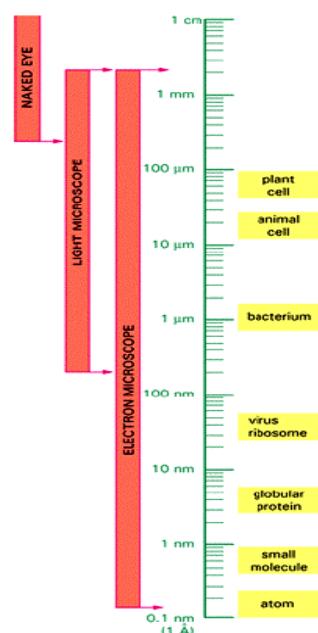


Abbildung 11:
Auflösungsvermögen [16]

1.2.2 3D-Bilder

1.2.2.1 Erstellung von 3D-Bildern

Die 3D-Bilder werden aus den Schnittbildern, die das konfokale Mikroskop (siehe Punkt 1.2.1.1) liefert, berechnet. Diese Berechnung wird mittels der kommerziellen Software Amira [18] durchgeführt und das Ergebnis wird im VRML-Format abgespeichert.

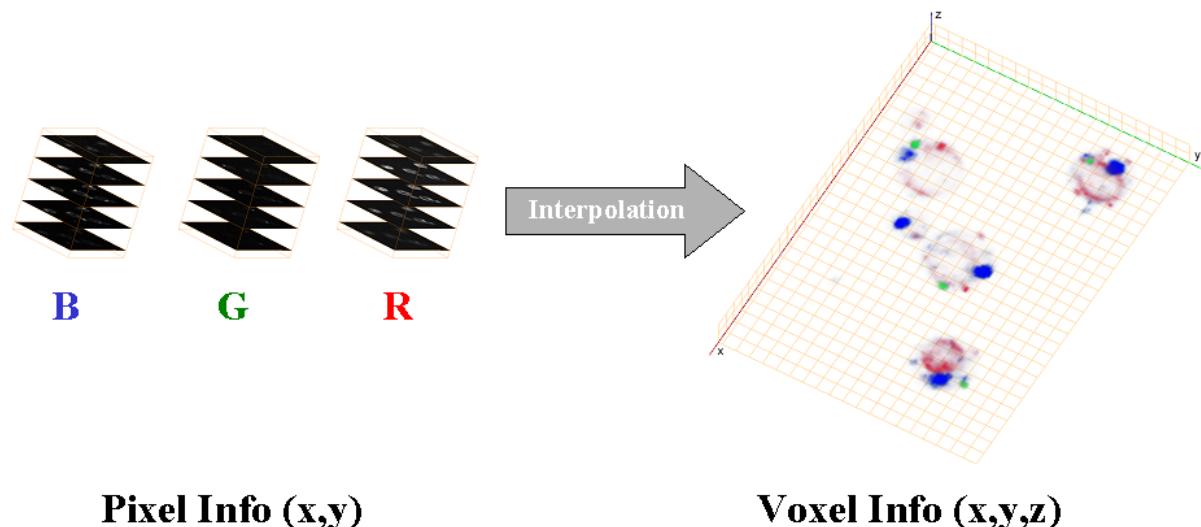


Abbildung 12: Prinzip der Voxel Generierung

Am IMBM der Uni Graz steht derzeit das konfokal Lasermikroskop Leica TCS SP2 AOBS [9] zur Verfügung. Laut Herstellerangaben hat es ein Auflösungsvermögen in der X/Y-Ebene von 150nm und in der Z-Achse 320nm. Unter Berücksichtigung des Shannon Abtasttheorems verschlechtert sich das Auflösungsvermögen um den Faktor größer zwei. In der Praxis wird mit dem Faktor 2,3 gerechnet. Es ergibt sich daraus ein reales Auflösungsvermögen von 65nm in der X/Y-Ebene und von 140nm in der Z-Achse. Um bestmögliche Schnittbilder von einer Hefezelle ($\sim 5\mu\text{m}$ Durchmesser) zu erhalten, benötigt man folglich ca. 36 Schnitte mit einer Schnittweite von 140nm.

1.2.2.2 Anzeigen der 3D-Bilder

Die Firma Parallel Graphics bietet ein frei verfügbares VRML-Plugin an, welches unter <http://www.parallelgraphics.com/> herunter geladen werden kann. Dieses Plugin ermöglicht das Visualisieren der VRML-Dateien in den Webbrowsern Netscape, Mozilla und Microsoft Internet Explorer.

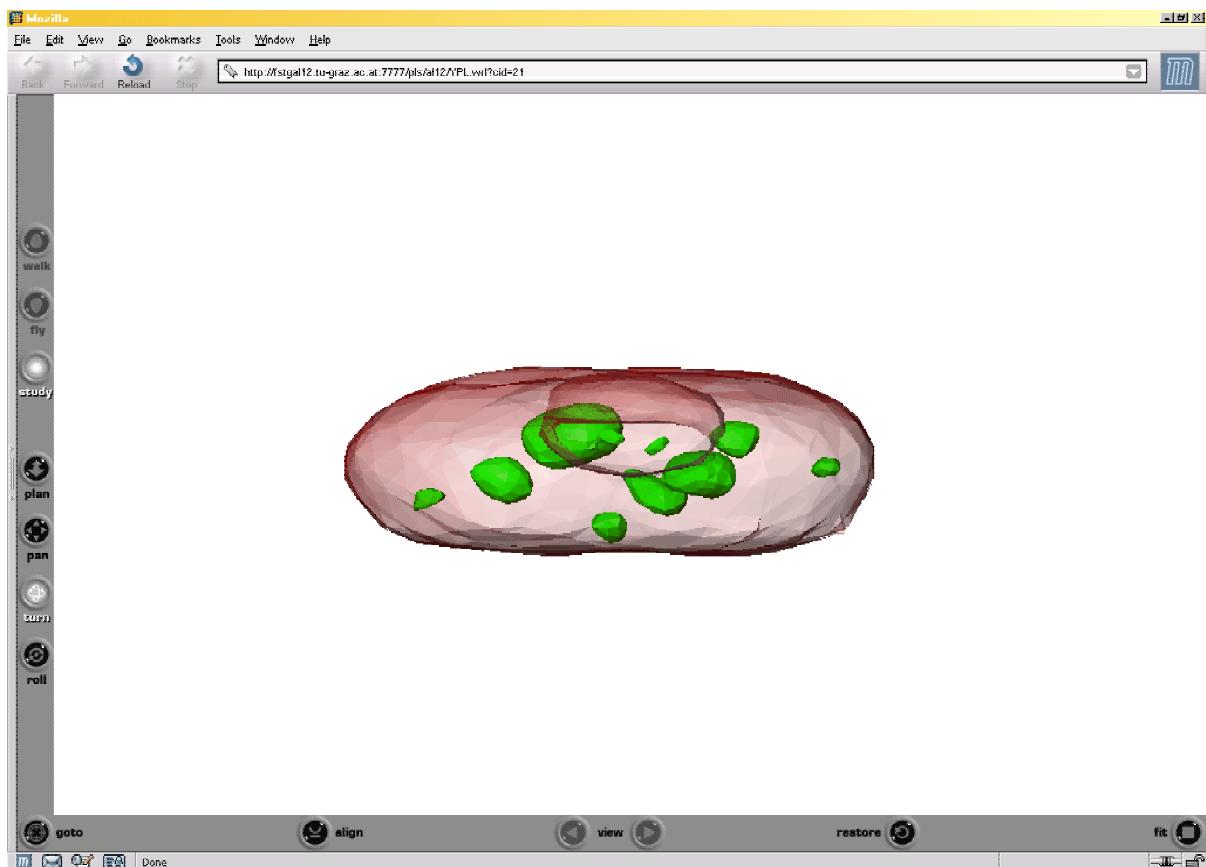


Abbildung 13: VRML-Plugin für Webbrower

1.2.3 Datenbanken

1.2.3.1 Einführung

Ein schneller Zugriff auf Informationen und deren Verwaltung spielt heute im sogenannten Informationszeitalter eine wesentliche Rolle. Wurden früher die Informationen auf Papier (analoge Form) abgelegt, so sind diese heute in digitaler Form verfügbar. Dies ermöglicht eine Beschleunigung und Optimierung der Verwaltung von Informationen. Des Weiteren werden durch Datenbanken Inkonsistenzen vermieden. Die Kehrseite ist jedoch eine immense Flut an Informationen, welche enormen Speicherplatz benötigen. [10]

Datenbanksysteme bestehen im wesentlichen aus zwei Teilen, der Datenbank (DB) und einem Datenbank Management System (DBMS). Die Datenbank ist ein strukturierter, von einem DBMS verwalteter Datenbestand, welcher auf einem oder mehreren Rechnern verteilt gespeichert wird. Das Datenbank Management System ist eine Software (Schicht), die zur Verwaltung von Datenbanken eingesetzt wird. Es darf nur über diese Software direkt auf den Datenbestand zugegriffen werden. [10]

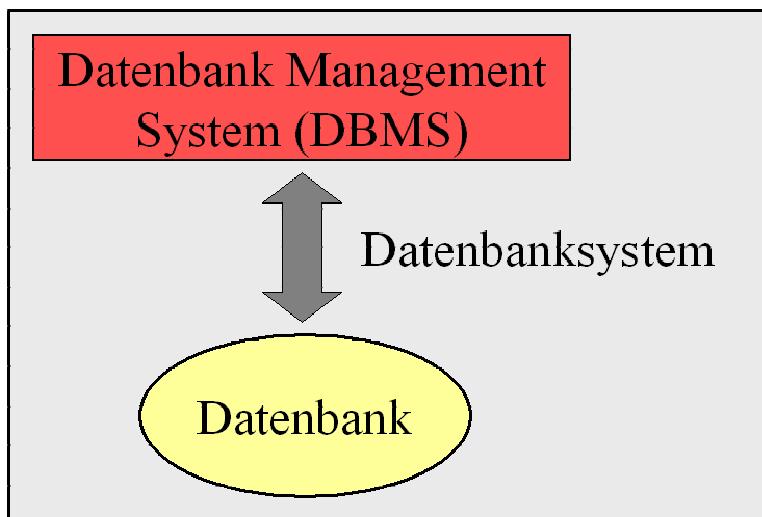


Abbildung 14: Datenbanksystem

Vorteile von Datenbanksystemen:

- Informationen sind zentral und widerspruchsfrei gesammelt und verwaltet.
- Informationen können miteinander verknüpft werden.
- Informationen können nach bestimmten Kriterien abgefragt werden.
- Informationen können in anderen Programmen verwendet werden.

1.2.3.2 Funktionen eines Datenbanksystems

Die relationale DBMS-Variante ist heute vorherrschend. Das relationale Datenbankmodell wurde von E. F. Codd [17] in 1970 begründet. Moderne Datenbanksysteme sollten folgende Basisfunktionalitäten besitzen:

- **Integration:** Die Datenintegration erfordert eine einheitliche Verwaltung aller von den Anwendungen benötigten Daten. Hier verbirgt sich die Möglichkeit der kontrollierten nicht-redundanten Datenhaltung des gesamten Datenbestandes.

- **Operationen:** Die Operationen oder Datenbankanweisungen zur Datenspeicherung, -suche, oder -änderung werden in einer einheitlichen Datenbanksprache angeboten. Die Operationen sind möglichst einfach in der Datenbanksprache formulierbar.
- **Katalog:** Die Daten können mit Hilfe von so genannten Katalogdaten oder auch Metadaten beschrieben werden. Die Katalogdaten können mit den Mitteln der Datenbanksprache ausgelesen werden.
- **Benutzersichten:** Für unterschiedliche Anwendungen sind unterschiedliche Sichten auf den Datenbestand notwendig. Dadurch können Benutzer nur Daten sehen, die für sie relevant sind. Die Definition von Sichten ist mit den Möglichkeiten des Datenschutzes stark verknüpft.
- **Konsistenzhaltung:** Die Konsistenzüberwachung oder Integritätssicherung übernimmt die Gewährleistung der Korrektheit von Datenbankinhalten und der korrekten Ausführung von Änderungen, sodass diese die Konsistenz nicht verletzen können.
- **Datenschutz:** Aufgabe des Datenschutzes ist der Ausschluss unautorisierter Zugriffe auf die gespeicherten Daten. Dies umfasst datenschutzrechtlich relevante Aspekte personenbezogener Informationen ebenso, wie den Schutz von firmenspezifischen Datenbeständen.
- **Transaktionen:** Folgen von Anweisungen zur Änderung von Daten lassen sich zu Funktionseinheiten zusammenfassen. Das Datenbanksystem garantiert entweder die vollständige Ausführung dieser Anweisungsfolgen, oder die Rücksetzung der Datenbank in den Zustand vor Beginn der Ausführung.
- **Synchronisation:** Transaktionen von Benutzern, die gleichzeitig mit dem Datenbanksystem arbeiten, werden synchronisiert. Dadurch werden Beeinflussungen, etwa versehentliche Schreibkonflikte auf gemeinsam benötigte Daten, vermieden.
- **Datensicherung:** Die Daten werden nach Systemfehlern, etwa Plattenfehlern, wieder in einen konsistenten Zustand überführt. Dieses Rücksetzen der Datenbank sollte weitgehend automatisch geschehen. [10]

1.2.3.3 Einsatz von Datenbanksystemen

Neben den Vorteilen, die Datenbanksysteme gegenüber den Dateisystemen bieten, gibt es auch Gründe, die eher für den Einsatz von Dateisystemen sprechen. Dazu zählen die mit der Anschaffung verbundenen direkten Kosten, der Aufwand an Einarbeitungszeit und die notwendige laufende Betreuung.

Aufgrund der höheren Leistung, verglichen mit der eines Dateisystems, verursachen Datenbanksysteme darüber hinaus auch im Betrieb mehr Kosten, die sich vor allem durch eine größere Rechnerbelastung bemerkbar machen. Diese höheren Kosten müssen gegen die gebotenen Mehrleistungen abgewogen werden. [10]

EINSATZ EINES DATENBANKSYSTEMS ODER EINES DATEISYSTEMS	
Einsatz eines Datenbanksystems	Einsatz eines Dateisystems
Umfangreiche Datenmengen sind vorhanden, die von mehreren Anwendungsprogrammen bearbeitet werden	Die Datenbasis und die Anwendungen sind sehr einfach, und es sind keine Änderungen geplant.
Eine neue umfangreiche Datensammlung ist geplant, für diese Daten sind zur Zeit aber noch keine vielfältigen Anwendungsprogramme abzusehen.	Die Antwortzeiten der Anwendungsprogramme müssen immer extrem kurz sein (Real-Time-Anwendungen).
Es existiert bereits eine kleinere Dateisammlung, und es werden in Zukunft laufend weitere Dateien hinzukommen.	Kein Mehrbenutzerzugriff zu den Daten ist nötig.

Tabelle 3: Einsatz eines Datenbanksystems oder eines Dateisystems [10]

1.3 Aufgabenstellung

Die bereits vorhandene YPL.db [7] soll nach hier aufgeführten Kriterien erweitert bzw. modifiziert werden. Die Aufgabenstellung umfasst zwei Bereiche: Ein Bereich beinhaltet die Forms Masken, welche nur für interne Benutzer zugänglich sind. Der zweite Bereich beinhaltet das öffentlich zugängliche Webinterface.

Kriterien/Aufgaben für Forms Masken:

- Für die Experimente soll ein Feld „construct verification“ eingeführt werden.
- Des Weiteren soll es möglich sein zu den jeweiligen Experimenten 3D-Bilder in die Datenbank zu laden.
- Bei der Erstellung von Webimages (Bilder die öffentlich zugänglich sind) soll es möglich sein Gruppen zu erstellen und diesen Webimages zuzuweisen. Durch eine Freigabe der Gruppe werden alle Webimages, die dieser angehören, frei gegeben.
- Die Experimente, welche über das Webinterface eingetragen wurden, sind mittels einer Web-Import-Maske in YPL.db zu integrieren. Datensätze bzw. Bilder werden vor dem Übertragen visualisiert und können gegebenenfalls noch geändert werden.
- Vorhandene Funktionsfehler in YPL.db sind gegebenenfalls zu beheben.

Kriterien/Aufgaben für Webinterface:

- Ein öffentlich zugängliches Webinterface soll erstellt werden, mit dessen Hilfe ganze Experimente und deren Daten einfach und rasch eingegeben werden können. Nach deren Überprüfung können diese direkt in die YPL.db eingespielt werden.
- Ein Online-Manual für das YPL.db Webinterface soll erstellt werden.
- Zur Steigerung der Usability sind die Abfrageergebnisse experimentweise und tabellarisch anzuzeigen. Zu jeden gefundenen Experiment kann eine Bilder-Gallery angezeigt werden.
- Für eine bessere Auswertung der Bilder sollten sie mit einstellbarer Transparenz übereinander gelegt werden können.
- Falls ein 3D-Bild zu einem Experiment vorhanden ist, so ist dies dem Benutzer mittels eingeblendeten Link mitzuteilen.

2 Methoden

In diesem Kapitel wird auf Konzepte, Programmiersprachen, Entwicklungswerkzeuge und auf Lösungswege eingegangen.

2.1 Datenbanken

2.1.1 Das relationale Datenmodell

1970 veröffentlichte Dr. E. F. Codd [17] in der renommierten Zeitschrift „Communications of the ACM“ einen Aufsatz mit dem Titel „A Relational Model of Data for Large Shared Data Banks“. Dieser Artikel stellte die theoretische und mathematische Grundlage für das Konzept des relationalen Datenmodells dar, in dem sowohl die klassischen Mengen-Operatoren als auch weitere Operatoren auf Relationen, die speziell auf die Bedürfnisse bei der Beschreibung und Manipulation von Daten zugeschnitten sind, enthalten sind.

Der Erfolg des relationalen Datenmodells beruht auf folgenden Punkten:

- Einfache Datenstruktur: Relation (vergleichbar mit einer Tabelle)
- mengenorientierte Verarbeitung der Daten
- wenig Grundoperationen zur Verarbeitung und dadurch eine klare Semantik
- formale Theorie zur Modellierung und Anfrageverarbeitung

Von den verschiedensten Anbietern werden DBMS angeboten, welche auf dem relationalen Datenmodell basieren. Zu den kommerziellen DBMS Anbietern gehören unter anderem Oracle, SQL Server, Sybase, Informix und DB2. Auch eine Reihe von nicht-kommerziellen Systemen stehen zur Verfügung, wie Interbase, mySQL, PostgreSQL und weitere.

STRUKTUR EINES RELATIONALEN MODELLS			
Datenbank			
Tabellen		Datensätze	
		Felder	

Tabelle 4: Struktur eines relationalen Modells

2.1.1.1 Einige Definitionen

- **Attribute:**

$$\mathbf{A} = \{ a_1, a_2, \dots, a_n \}$$

Endliche Menge von Attributen, wobei jedes Attribut $a \in \mathbf{A}$ einen nicht leeren Wertebereich (Domäne) $dom(a)$ atomarer Werte (z.B. Integer, String usw.) besitzt.

- **Relation:**

$$\mathbf{R} \subseteq dom(a_1) \times \dots \times dom(a_n)$$

Eine Teilmenge des kartesischen Produktes über den Attribut-Domains der zugrundeliegenden Attributmenge \mathbf{A} .

- **Tupel:**

Element einer Relation \mathbf{R} , welches genau einer Zeile, in der diese Relation darstellenden Tabelle, entspricht.

2.1.1.2 Relationenalgebra

ÜBERSICHT ÜBER OPERATOREN IN DER RELATIONENALGEBRA	
Operator	Ursprung
Produkt	Mengenlehre
Vereinigung	Mengenlehre
Differenz	Mengenlehre
Schnittmenge	Mengenlehre
Quotient	Mengenlehre
Projektion	Relational Theorie
Selektion	Relational Theorie
Verbund	Relational Theorie

Tabelle 5: Übersicht über Operatoren in der Relationalalgebra [11]

- **Produkt (Product):**

Das Ergebnis des Produktes zweier Relationen \mathbf{R} , \mathbf{S} umfasst alle Tupel, die aus allen möglichen Kombinationen beider Relationen erzeugt werden können.

$$\mathbf{R} \times \mathbf{S} := \{ (\alpha_1, \alpha_2, \dots, \alpha_m, \beta_1, \beta_2, \dots, \beta_n) \mid (\alpha_1, \alpha_2, \dots, \alpha_m) \in \mathbf{R}, (\beta_1, \beta_2, \dots, \beta_n) \in \mathbf{S} \}$$

$\begin{array}{ c c } \hline \alpha_1 & \beta_1 \\ \hline \alpha_2 & \beta_2 \\ \hline \alpha_3 & \beta_3 \\ \hline \end{array}$	\times	$\begin{array}{ c c } \hline \gamma_1 & \delta_1 \\ \hline \gamma_2 & \delta_2 \\ \hline \gamma_3 & \delta_3 \\ \hline \end{array}$	$=$	$\begin{array}{ c c c c } \hline \alpha_1 & \beta_1 & \gamma_1 & \delta_1 \\ \hline \alpha_1 & \beta_1 & \gamma_2 & \delta_2 \\ \hline \alpha_1 & \beta_1 & \gamma_3 & \delta_3 \\ \hline \alpha_2 & \beta_2 & \gamma_1 & \delta_1 \\ \hline \alpha_2 & \beta_2 & \gamma_2 & \delta_2 \\ \hline \alpha_2 & \beta_2 & \gamma_3 & \delta_3 \\ \hline \alpha_3 & \beta_3 & \gamma_1 & \delta_1 \\ \hline \alpha_3 & \beta_3 & \gamma_2 & \delta_2 \\ \hline \alpha_3 & \beta_3 & \gamma_3 & \delta_3 \\ \hline \end{array}$
--	----------	---	-----	--

- **Vereinigung (Union):**

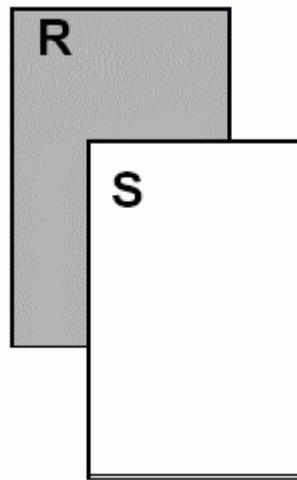
Das Ergebnis der Vereinigung zweier Relationen \mathbf{R} , \mathbf{S} umfasst alle Tupel aus beiden Relationen.

$$\mathbf{R} \cup \mathbf{S} := \{ \tau \mid \tau \in \mathbf{R} \vee \tau \in \mathbf{S} \}$$

$$\begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_2 & \beta_2 \\ \hline \alpha_3 & \beta_3 \\ \hline \end{array} \cup \begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_1 & \beta_4 \\ \hline \alpha_2 & \beta_3 \\ \hline \end{array} = \begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_1 & \beta_4 \\ \hline \alpha_2 & \beta_2 \\ \hline \alpha_2 & \beta_3 \\ \hline \alpha_3 & \beta_3 \\ \hline \end{array}$$

- **Differenz (Difference):**

Das Ergebnis der Differenz zweier Relationen \mathbf{R} , \mathbf{S} umfasst alle Tupel von \mathbf{R} , die nicht in \mathbf{S} enthalten sind.

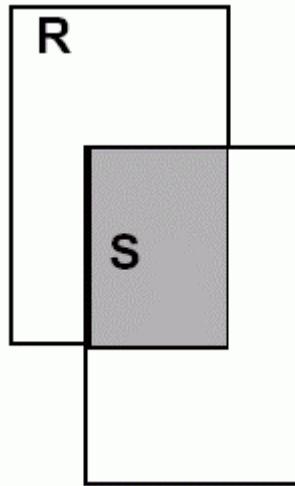


$$\mathbf{R} - \mathbf{S} := \{ \tau \in \mathbf{R} \mid \tau \notin \mathbf{S} \}$$

$$\begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_2 & \beta_2 \\ \hline \alpha_3 & \beta_3 \\ \hline \end{array} - \begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_1 & \beta_4 \\ \hline \alpha_2 & \beta_3 \\ \hline \end{array} = \begin{array}{|c|c|} \hline \alpha_2 & \beta_2 \\ \hline \alpha_3 & \beta_3 \\ \hline \end{array}$$

- **Schnittmenge (Intersection):**

Die Schnittmenge zweier Relationen \mathbf{R} , \mathbf{S} umfasst alle Tupel, die sowohl in \mathbf{R} als auch in \mathbf{S} vertreten sind.



$$\mathbf{R} \cap \mathbf{S} := \{ \tau \mid \tau \in \mathbf{R} \wedge \tau \in \mathbf{S} \}$$

$$\begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_2 & \beta_2 \\ \hline \alpha_3 & \beta_3 \\ \hline \end{array} \cap \begin{array}{|c|c|} \hline \alpha_1 & \beta_1 \\ \hline \alpha_1 & \beta_4 \\ \hline \alpha_2 & \beta_3 \\ \hline \end{array} = \boxed{\alpha_1 \quad \beta_1}$$

- **Quotient (Division):**

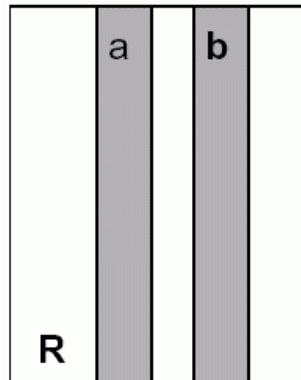
Der Quotient zweier Relationen $\mathbf{R}(A)$, $\mathbf{S}(B)$, wobei A und B eine nichtleere Schnittmenge X von Attributen besitzen, enthält alle Tupel von \mathbf{R} (eingeschränkt auf $A-X$), die mit allen Tupeln in \mathbf{S} in Relation stehen.

$$\mathbf{R \% S} := \{ (\alpha_{k+1}, \dots, \alpha_m) \mid \forall (\beta_1, \dots, \beta_n) \in \mathbf{S} : (\beta_1, \dots, \beta_k, \alpha_{k+1}, \dots, \alpha_m) \in \mathbf{R} \}$$

$$\begin{array}{|c|c|} \hline a & b \\ \hline \alpha_1 & \beta_1 \\ \hline \alpha_1 & \beta_2 \\ \hline \alpha_1 & \beta_3 \\ \hline \alpha_2 & \beta_1 \\ \hline \alpha_2 & \beta_3 \\ \hline \alpha_3 & \beta_1 \\ \hline \alpha_3 & \beta_2 \\ \hline \end{array} \stackrel{\mathbf{R \% S}}{=} \begin{array}{|c|} \hline b \\ \hline \beta_1 \\ \hline \beta_2 \\ \hline \beta_3 \\ \hline \end{array} = \boxed{\begin{array}{|c|} \hline a \\ \hline \alpha_1 \\ \hline \end{array}}$$

- **Projektion (Projection):**

Die Projektion einer Relation \mathbf{R} bzgl. einer Teilmenge der Attribute umfasst alle Tupel, die nach dem Entfernen der übrigen Attribute (Spalten) in der nunmehr verkürzten Relation verbleiben.



$$\pi_B(\mathbf{R}) := \{ \tau[B] \mid \tau \in \mathbf{R} \}$$

$\tau[B]$... τ eingeschränkt auf B

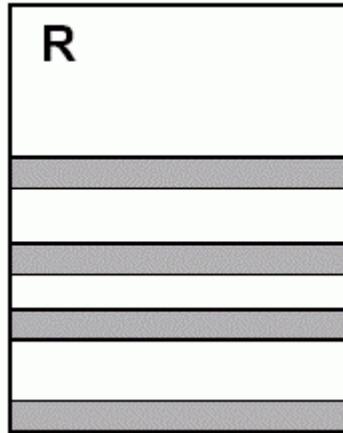
$$\pi_{a,b,c} =$$

a	b	c	d
α_1	β_1	γ_1	δ_1
α_1	β_1	γ_2	δ_2
α_1	β_1	γ_3	δ_3
α_2	β_2	γ_1	δ_1
α_2	β_2	γ_2	δ_2
α_2	β_2	γ_3	δ_3
α_3	β_3	γ_1	δ_1
α_3	β_3	γ_2	δ_2
α_3	β_3	γ_3	δ_3

a	b	c
α_1	β_1	γ_1
α_1	β_1	γ_2
α_1	β_1	γ_3
α_2	β_2	γ_1
α_2	β_2	γ_2
α_2	β_2	γ_3
α_3	β_3	γ_1
α_3	β_3	γ_2
α_3	β_3	γ_3

- **Selektion (Selection):**

Das Ergebnis einer Selektion der Relation \mathbf{R} , bezüglich einer Bedingung C , umfasst alle Tupel, die die Bedingung C erfüllen.



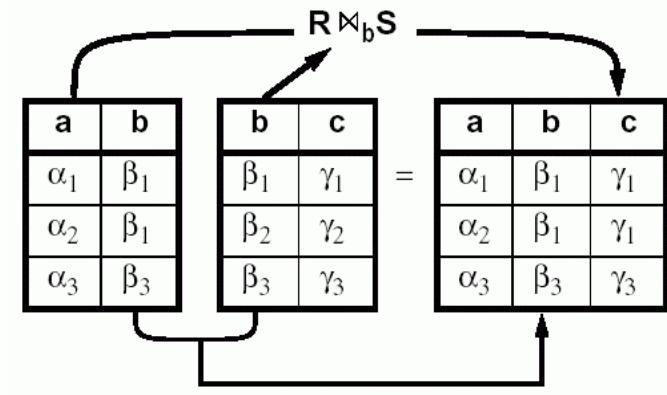
$$\sigma_C(R) := \{ \tau \in R \mid \tau \text{ erfüllt } C \}$$

$$\sigma_{b=\beta_2} \quad \begin{array}{|c|c|c|c|} \hline a & b & c & d \\ \hline \alpha_1 & \beta_1 & \gamma_1 & \delta_1 \\ \hline \alpha_1 & \beta_1 & \gamma_2 & \delta_2 \\ \hline \alpha_1 & \beta_1 & \gamma_3 & \delta_3 \\ \hline \alpha_2 & \beta_2 & \gamma_1 & \delta_1 \\ \hline \alpha_2 & \beta_2 & \gamma_2 & \delta_2 \\ \hline \alpha_2 & \beta_2 & \gamma_3 & \delta_3 \\ \hline \alpha_3 & \beta_3 & \gamma_1 & \delta_1 \\ \hline \alpha_3 & \beta_3 & \gamma_2 & \delta_2 \\ \hline \alpha_3 & \beta_3 & \gamma_3 & \delta_3 \\ \hline \end{array} = \begin{array}{|c|c|c|c|} \hline a & b & c & d \\ \hline \alpha_2 & \beta_2 & \gamma_1 & \delta_1 \\ \hline \alpha_2 & \beta_2 & \gamma_2 & \delta_2 \\ \hline \alpha_2 & \beta_2 & \gamma_3 & \delta_3 \\ \hline \end{array}$$

- **Natürlicher Verbund (Natural Join):**

Das Ergebnis eines natürlichen Verbundes der Relationen \mathbf{R} , \mathbf{S} über Attribute a von \mathbf{R} und b von \mathbf{S} enthält das kartesische Produkt aller Tupel von \mathbf{R} , \mathbf{S} , die in a bzw. b die gleichen Werte enthalten, wobei die "doppelte Spalte" nur einmal auftaucht.

$$\mathbf{R} \bowtie_{a=b} \mathbf{S} := \{ (\alpha_1, \alpha_2, \dots, \alpha_m, \beta, \beta_1, \beta_2, \dots, \beta_n) \mid (\alpha_1, \alpha_2, \dots, \alpha_m, \beta) \in \mathbf{R}, (\beta, \beta_1, \beta_2, \dots, \beta_n) \in \mathbf{S} \}$$



2.1.2 Datenintegrität

Das Konzept der Datenintegrität besagt, dass in einer Tabelle Tupel nicht doppelt bzw. mehrfach vorkommen dürfen. Infolgedessen muss eine Tabelle einen Primärschlüssel, laut der relationalen Theorie, besitzen. [12]

2.1.2.1 Primärschlüssel

Der Primärschlüssel einer Tabelle besteht aus einer Gruppe von Attributen, die einen Datensatz (Zeile) eindeutig identifizieren. Er kann aus einem oder mehreren Attributen bestehen, wobei die Attribute keine(n) NULL-Wert(e) haben dürfen. Nicht immer ist es leicht ersichtlich aus welchen Attributen sich der Primärschlüssel zusammensetzt. Dies kann jedoch bei der Durchsicht der Daten am einfachsten ermittelt werden. [12]

2.1.2.2 Referentielle Integrität

Tabellen werden zueinander mit Hilfe von Fremdschlüsseln in Beziehung gebracht. Ein Fremdschlüssel besteht aus einer oder mehreren Spalten, deren Werte im Primärschlüssel einer andern Tabelle wiederzufinden sind. Die referentielle Integrität ist erfüllt, wenn die Werte in der Spalte eines Fremdschlüssels auf den referenzierten Primärschlüssel oder auf den NULL-Wert eingeschränkt sind. Nach der Deklaration der Primär- und Fremdschlüssel ist die Erhaltung der referentiellen Integrität Aufgabe des RDBMS (Relational Database Management System). [12]

2.1.2.3 Beziehungen

Unter einer Beziehung versteht man die Verbindung zwischen zwei Tabellen. Hierbei wird eine Tabelle als übergeordnet, die andere als untergeordnet bezeichnet. [12]

2.1.2.4 Normalisierung

Unter der Normalisierungstheorie versteht man die Untersuchung von Tabellen, Attributen und der Abhängigkeit der Attribute zueinander.

Ziele der Normalisierung:

- Minimierung redundanter Daten
- Vermeiden von Änderungsanomalien
- Verhindern inkonsistenter Daten
- Entwerfen von Datenstrukturen, die eine einfache Pflege erlauben

Es werden hier nur die ersten drei Normalformen behandelt, da die vierte und fünfte Normalform komplexere Normalisierungsaspekte behandeln und den Rahmen sprengen würden. Im Zusammenhang zur Normalisierungstheorie werden in Unterlagen, zum Thema Datenbanken, einige Begriffe synonym verwendet. Siehe Tabelle 6: Übersicht über Datenbankterminologie [12].

ÜBERSICHT ÜBER DATENBANKTERMINOLOGIE		
Theoretiker	Analytiker	Entwickler
Relation	Entity	Tabelle
Attribut	Attribut	Spalte
Tupel	Zeile	Zeile/Datensatz

Tabelle 6: Übersicht über Datenbankterminologie [12]

- **Erste Normalform:**

Die erste Normalform ist gegeben, wenn alle Attribute einer Relation in atomarer Form vorliegen. Das heißt, jedes Attribut aus einer einzigen Information über die Relation besteht.

Kurz: Jede Spalte sollte genau einen Teil der Information enthalten.

- **Zweite Normalform:**

Die zweite Normalform ist gegeben, wenn alle Attribute einer Relation nur vom Primärschlüssel abhängen. Eine Relation darf keine Fremdschlüssel enthalten.

Kurz: Alle Tupel hängen nur vom Primärschlüssel ab.

- **Dritte Normalform:**

Die dritte Normalform ist gegeben, wenn alle Attribute einer Relation vollständig vom Primärschlüssel abhängen. Unter vollständig versteht man, dass jedes Attribut der Tabelle vom gesamten Primärschlüssel abhängt.

Kurz: Alle Spalten hängen einzig und allein vom Primärschlüssel ab.

2.1.2.5 Transaktionen und operationale Integrität

Bei gleichzeitiger bzw. quasiprallen Nutzung der Datenbank durch mehrere Benutzer ergeben sich Integritätsprobleme. Um unerwünschte Effekte zu vermeiden, müssen Aktivitäten auf gemeinsame Daten geeignet synchronisiert werden. Überdies muss bei einem physikalischen Speicherfehler die Konsistenz der Daten gegeben bleiben. Das Konzept der Transaktion beruht auf der Sicherstellung der operationalen Integrität und ist in fast allen verfügbaren Datenbanken implementiert. Eine Transaktion besteht aus einer Sequenz von DDL (Data Definition Language; siehe Punkt 2.1.4) und/oder DML (Data Manipulation Language; siehe Punkt 2.1.5) Operationen.

Folgende Punkte sind einzuhalten:

- **Atomarität:**

Eine Transaktion kann nur vollständig abgeschlossen werden (Commit) oder vollständig verworfen werden (Rollback).

- **Konsistenz:**

Nach dem Abschluss der Transaktion ist die Datenbank in einem konsistenten Zustand.

- **Isolation:**

Eine Transaktion wird erst nach deren Abschluss für andere Benutzer sichtbar, und ist erst dann wieder durch eine neue Transaktion manipulierbar.

- **Dauerhaftigkeit:**

Die Änderungen in der Datenbank sind nach einer Transaktion dauerhaft festgeschrieben.

2.1.3 Entity-Relationship Modell

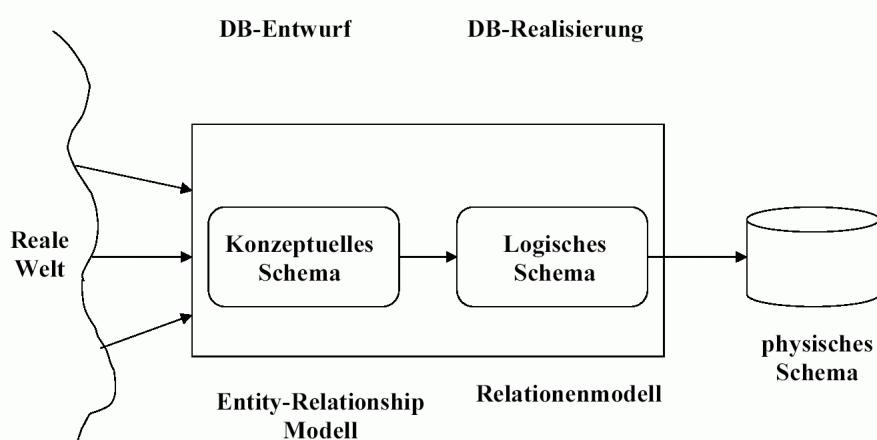


Abbildung 15: Datenbankentwurf mit ER-Modell [13]

Das von Chen (1976) [23] vorgestellte ER-Modell gilt als Standard Modellierungs-technik im Datenbankbereich. Die Aufgabe des ER-Modell besteht darin, die komplexe Wirklichkeit in ein Logisches Schema zu bringen. Das Entity-Relationship Modell ist unabhängig von hierarchischen, netzartigen und relationalen Datenmodell. Die wesentlichen Darstellungselemente des ER-Modelles sind Entities (Objekte) und Relationships (Beziehungen) zu ihnen.

2.1.4 Data Definition Language (DDL)

Unter DDL versteht man eine Sprache, die es ermöglicht die Struktur und Instanzen einer Datenbank zu definieren. Unter SQL stehen hierfür die Operationen Create (Erstellen), Alter (Ändern) und Drop (Löschen) zur Verfügung.

2.1.5 Data Manipulation Language (DML)

Unter DML versteht man eine Sprache, die es ermöglicht Daten in einer Datenbank zu manipulieren. SQL stellt hierfür die Statements Insert (Einfügen), Update (Überschreiben) und Delete (Löschen) zur Verfügung.

2.2 OAS (Oracle Application Server)

Der OAS stellt die Eigenschaften und Funktionen der Oracle Datenbank als webbasierten Service zur Verfügung. Der Client kann mittels HTTP-Browser auf diesen Web-Service zugegriffen.

2.2.1 Prinzip

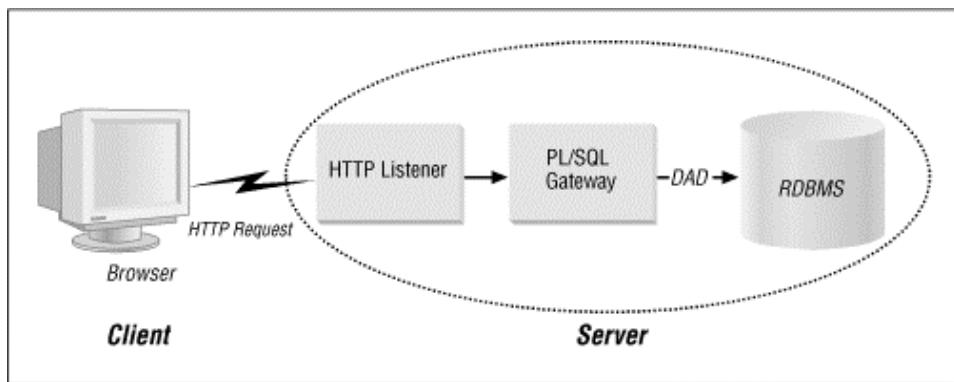


Abbildung 16: Übersicht über OAS [14]

Der HTTP-Listener empfängt die Anforderungen (Requests) der Clients (Webbrowser), diese werden in Path-Sections aufgelöst (URL Resolution). Entspricht die Path-Section einem virtuellen Verzeichnis, so handelt es sich um eine statische Ressource und der Listener gibt die angeforderte Ressource dem Client zurück. Wenn der Path einem DAD (Database Access Descriptor) entspricht, so handelt es sich um eine dynamische Ressource und die Anforderung wird dem PL/SQL-Gateway weitergeleitet. Das PL/SQL-Gateway zerlegt die URL nochmals, es wird der DAD Name extrahiert und die Namen der Prozeduren die ausgeführt werden sollen. Überdies werden auch die Parameter extrahiert. Die Prozedur wird mit ihren Parametern direkt in der Datenbank ausgeführt. Meist werden Prozeduren aus dem Paket PL/SQL-TOOLKIT aufgerufen. Nach dem Ausführen der Prozedur signalisiert das Gateway dem HTTP-Listener, dass die HTML Ausgabe der Prozedur dem Client-Browser gesendet wird. Der Client-Browser zeigt die Seite, wie eine normale statische Ressource an.

Die URL ist folgendermaßen aufgebaut:

`http://server[:port]/agent_name/plsql/[package.]procedure[?parm1=foo...]`

2.2.2 PL/SQL-TOOLKIT

Das PL/SQL-TOOLKIT ist ein Satz von PL/SQL Paketen, die von Oracle, zur Entwicklung von Web-Anwendungen, zur Verfügung gestellt werden. Diese Pakete generieren HTML dynamisch, führen Textoperationen aus und steigern die Entwicklungsproduktivität. Unter Tabelle 7: PL/SQL-Toolkit Pakete sind die enthaltenen Pakete aufgelistet.

PL/SQL-TOOLKIT PAKETE	
Paketname	Verwendung
HTF	HTML Parsen
HTP	HTML Generieren
OWA_COOKIE	Speichern der Cookies
OWA_OPT_LOCK	Records sperren
OWA_PATTERN	Suche und ersetzen von Text
OWA_SEC	Sicherheitsverwaltung
OWA_TEXT	Überlange Strings
OWA_UTIL	Steigerung der Produktivität

Tabelle 7: PL/SQL-Toolkit Pakete [14]

Eine PL/SQL Prozedur Hello World würde folgendermaßen aussehen:

```
CREATE OR REPLACE PROCEDURE hello_world
IS
BEGIN
    HTP.htmlopen;
    HTP.headopen;
    HTP.title ('You knew it was coming... ');
    HTP.headclose;
    HTP.comment ('This phrase is in every computer book.');
    HTP.bodyopen (cattributes => 'body bgcolor=blue');
    HTP.print ('And here it is .... Hello, World!');
    HTP.bodyclose;
    HTP.htmlclose;
END;
```

2.3 Programmierwerkzeuge

Es sind hier alle Programmierwerkzeuge angeführt, die zur Erstellung von YPL.db II erforderlich waren.

2.3.1 Oracle Developer Suite

Die Oracle Developer Suite ist eine Suite integrierter Werkzeuge für den Entwurf und die Erstellung von Anwendungen. Grafisch orientierte Anwendungen für Windows- Mac- und UNIX-Motife-Umgebungen können damit entwickelt werden.

Die Developer Suite besteht aus folgenden Komponenten:

- Oracle Forms
- Oracle Reports
- Oracle Graphics
- Procedure Builder
- Schema Builder
- Query Builder
- Translation Builder

2.3.1.1 Forms Builder

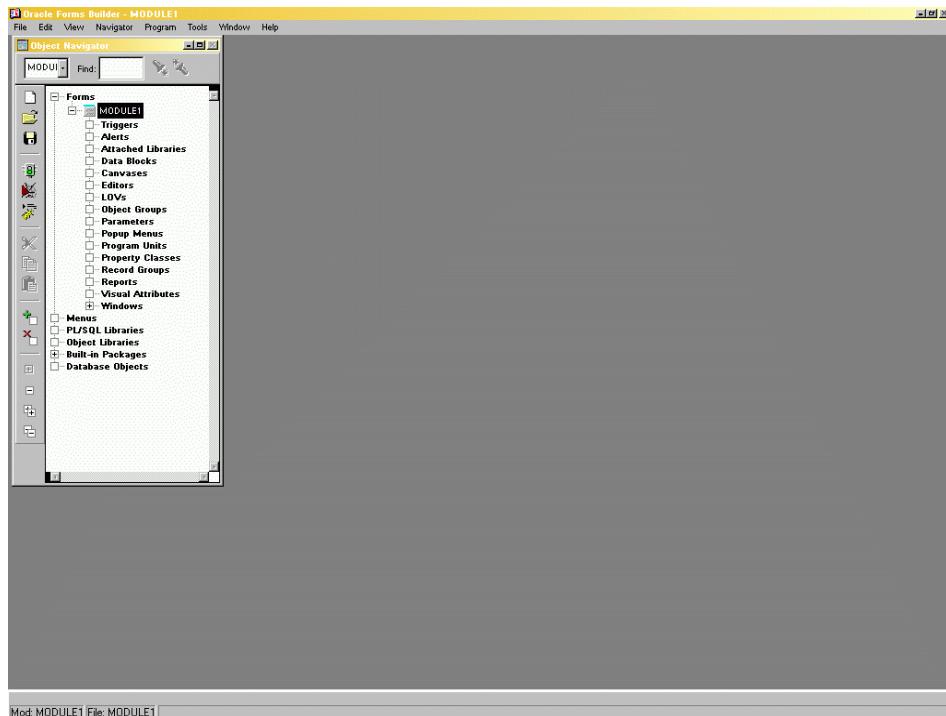
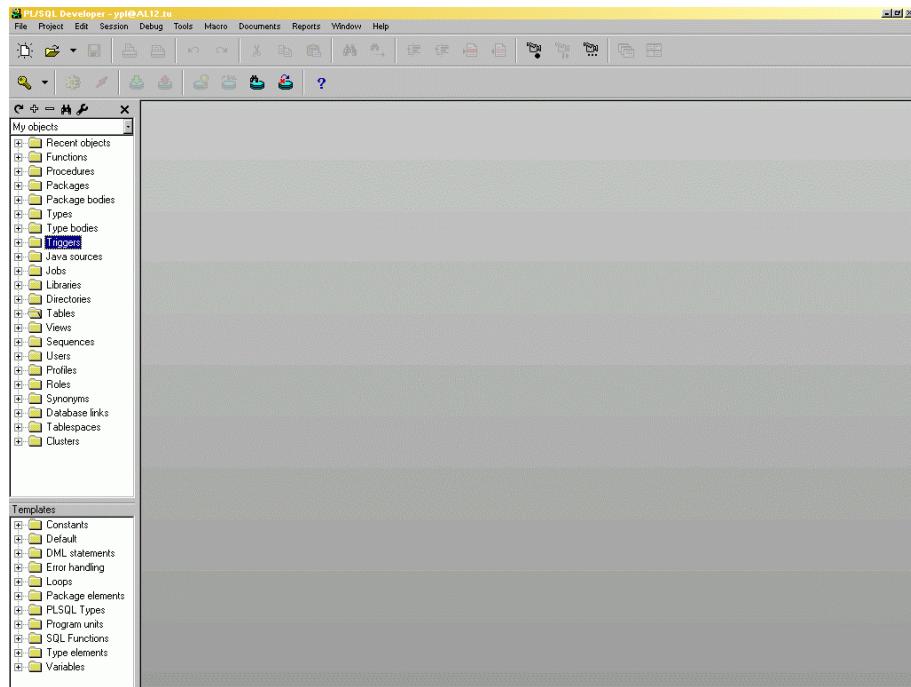


Abbildung 17: GUI des Oracle Form Builder

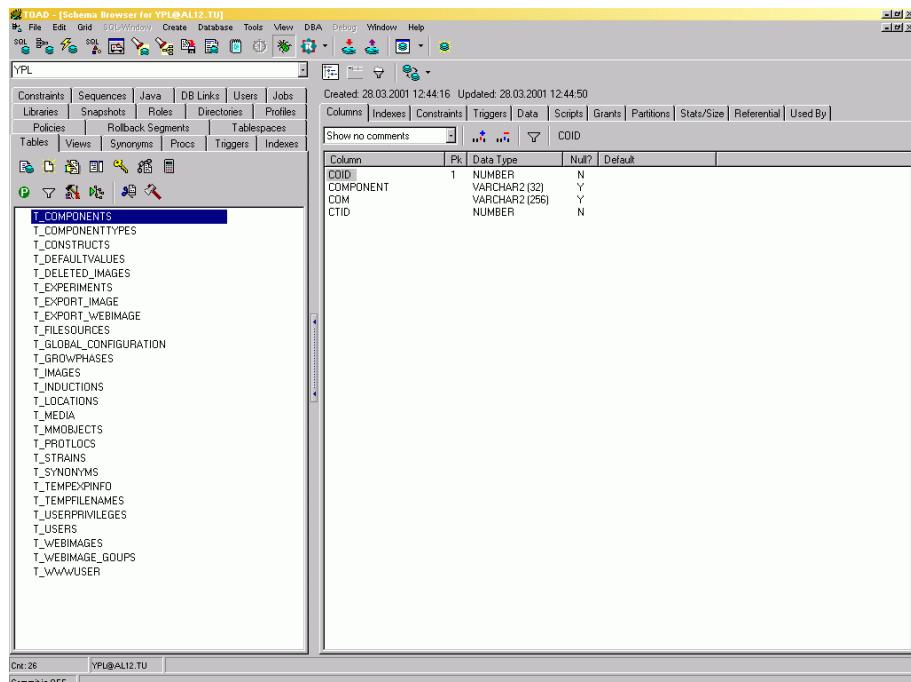
Oracle Forms ist eine GUI Entwicklungsumgebung mit robusten Datenauswertungsfähigkeiten. Es können mit Forms Formulare entwickelt werden, die Daten in der Datenbank einfügen, aktualisieren und löschen. Forms verwendet Client-Server Architektur, die Kommunikation mit dem Datenbank-Server erfolgt über Oracle SQL*Net.

2.3.2 PL/SQL Developer



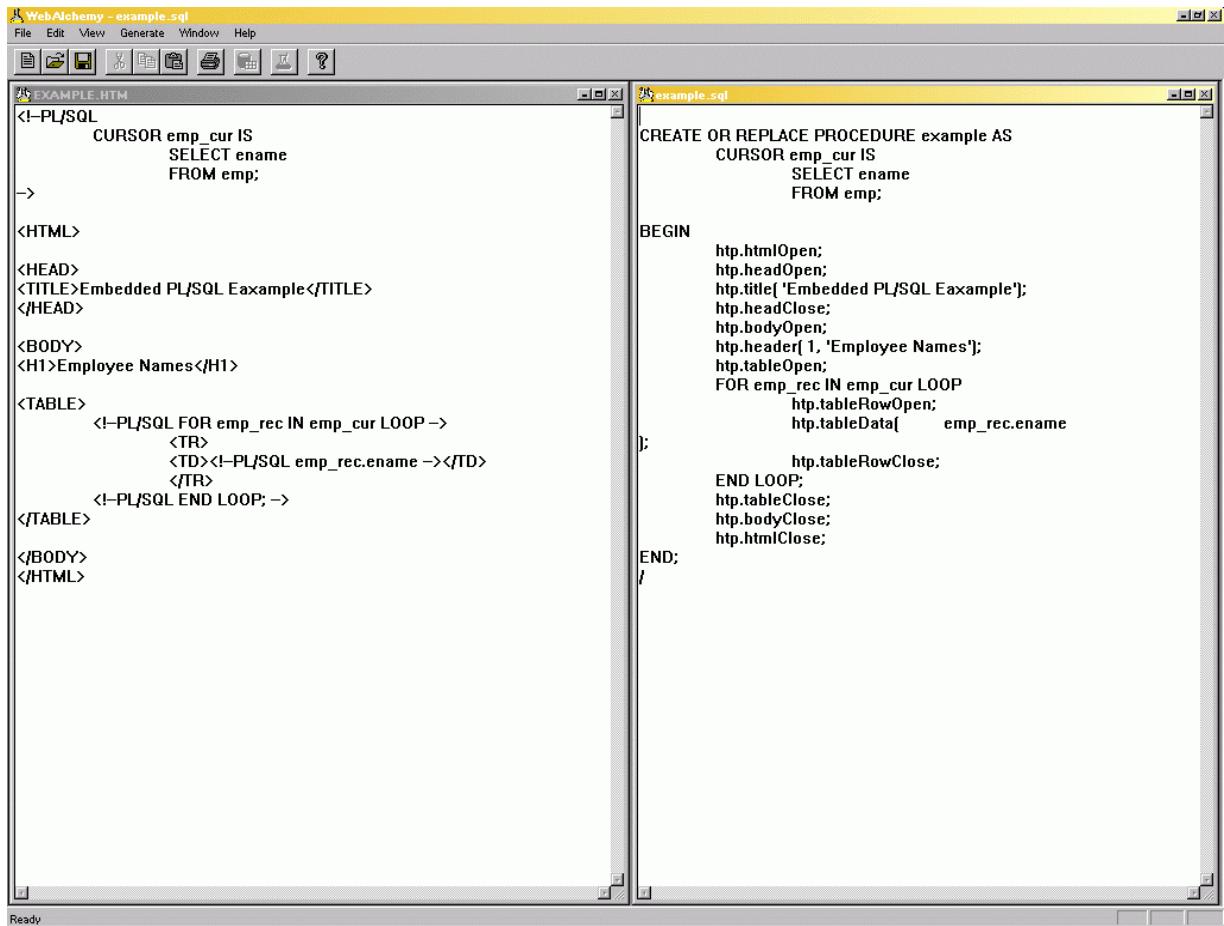
PL/SQL Developer ist eine Integrierte Entwicklungs Umgebung (IDE) für PL/SQL Anwendungsentwicklung in Oracle Datenbank. [20]

2.3.3 TOAD



TOAD ist ein Industriestandard Tool für PL/SQL Anwendungsentwicklung für Oracle Datenbanken von der Firma Quest Software. Bietet in etwa den selben Funktionsumfang wie das Programm PL/SQL Developer (siehe Punkt 2.3.2). [19]

2.3.4 WebAlchemy



The screenshot shows the WebAlchemy interface with two windows side-by-side. The left window, titled 'EXAMPLE.HTM', contains the following PL/SQL code:

```
<!--PL/SQL
CURSOR emp_cur IS
  SELECT ename
  FROM emp;
-->

<HTML>
<HEAD>
<TITLE>Embedded PL/SQL Eaxample</TITLE>
</HEAD>

<BODY>
<H1>Employee Names</H1>
<TABLE>
  <!--PL/SQL FOR emp_rec IN emp_cur LOOP -->
    <TR>
      <TD><!--PL/SQL emp_rec.ename --></TD>
    </TR>
  <!--PL/SQL END LOOP; -->
</TABLE>
</BODY>
</HTML>
```

The right window, titled 'example.sql', contains the generated PL/SQL code:

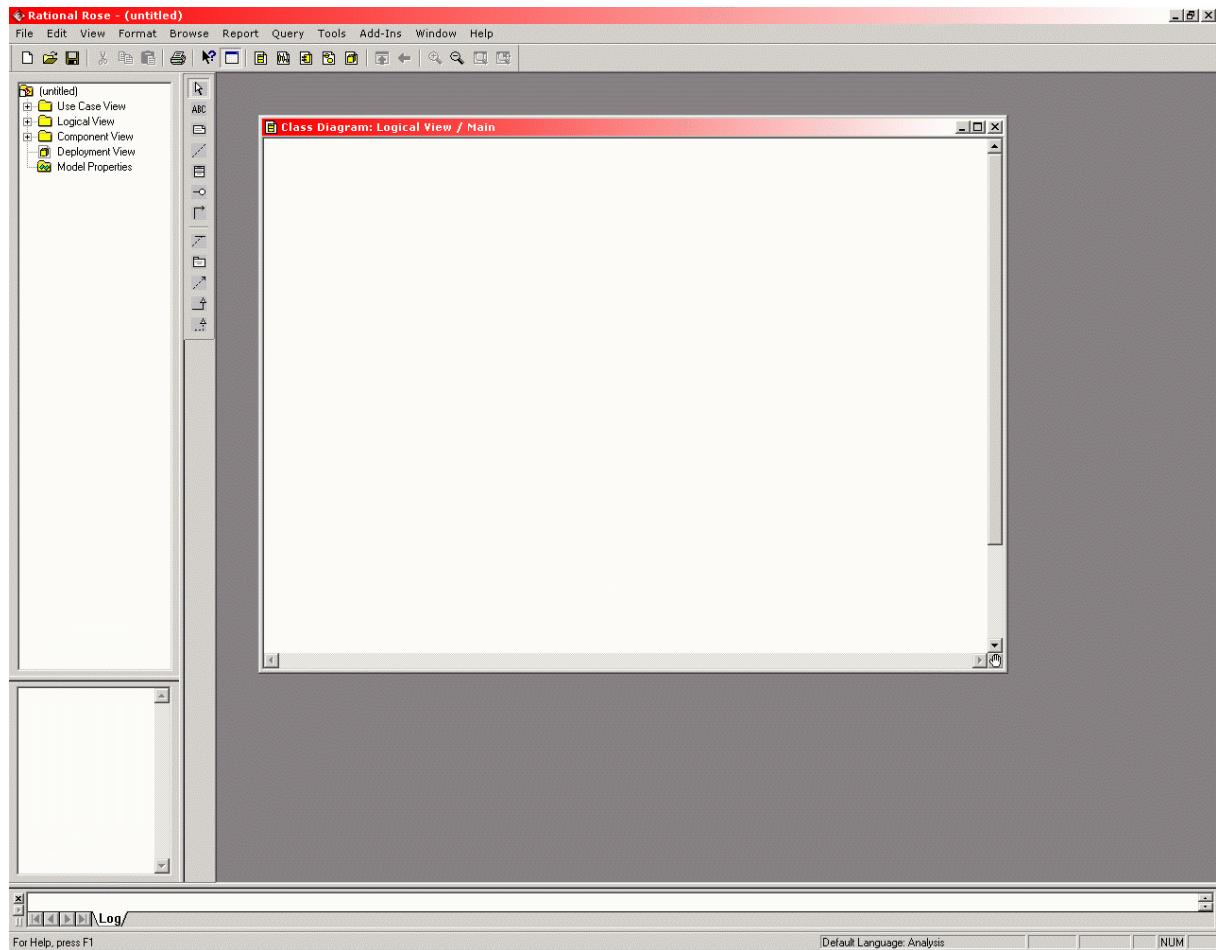
```
CREATE OR REPLACE PROCEDURE example AS
  CURSOR emp_cur IS
    SELECT ename
    FROM emp;

BEGIN
  htp.htmlOpen;
  htp.headOpen;
  htp.title('Embedded PL/SQL Eaxample');
  htp.headClose;
  htp.bodyOpen;
  htp.header(1, 'Employee Names');
  htp.tableOpen;
  FOR emp_rec IN emp_cur LOOP
    htp.tableRowOpen;
    htp.tableData( emp_rec.ename
  );
  htp.tableRowClose;
END LOOP;
htp.tableClose;
htp.bodyClose;
htp.htmlClose;
END;
/
```

WebAlchemy ist ein GUI-Werkzeug, mit dessen Hilfe HTML-Seiten in PL/SQL-Prozeduren umgewandelt werden können. Die Erstellung von PL/SQL-Prozeduren wird hierbei erheblich erleichtert, da das Layout in einem gewöhnlichen HTML-Editor erstellt werden kann, die von WebAlchemy in PL/SQL-Prozeduren umgewandelt. Es muss nur mehr die PL/SQL-Logik eingebaut werden.

WebAlchemy wurde von einem Oracle Consultant in Australien entwickelt und ist frei im Internet verfügbar. [21]

2.3.5 Rational Rose



Rational Rose ist das weltweit führende Werkzeug für die visuelle Modellierung und ermöglicht die Darstellung der Software in UML (Unified Modelling Language) Notation.

2.4 Usability

2.4.1 Kriterien

Für eine gute Usability sollten folgende Kriterien erfüllt sein [24]:

- **Einfachheit:**
 - leicht zu überblicken
 - leicht zu lernen
 - keine überflüssige Funktionalität
- **Vorhersehbarkeit:**
 - erwartungskonform
 - konsistent
 - logische Struktur
- **Transparenz:**
 - wo befinde ich mich
 - welche Möglichkeiten habe ich
- **Effizienz:**
 - minimaler Aufwand um Aktionen zu setzen
- **Fehlertoleranz:**
 - Design muss Fehleingaben des Anwenders minimieren
 - Möglichkeit Fehleingaben rückgängig zu machen

In der Praxis können leider nicht immer alle Kriterien eingehalten werden. Von Fall zu Fall muss hier abgewogen werden, inwieweit diese Kriterien verletzt werden können. In YPL.db II wurde speziell bei den öffentlich zugänglichen Webinterfaces (siehe Punkt 3.3.1 und Punkt 3.3.2) großes Augenmerk auf ihre Usability gelegt und bei den internen administrativen Bereichen (siehe Punkt 3.2) mehr auf deren Funktionalität gesetzt.

2.4.2 Anzahl der Tests

Jakob Nielson und Thomas Landauer [15] haben bei ihren Untersuchungen festgestellt, dass eine Anzahl von 5 Personen für Usability Tests vollkommen ausreichend ist. Die gefundenen Usability-Probleme folgen der Gleichung $N(1-(1-L)^n)$ (siehe Abbildung 18: Gefundene Usability-Probleme), wobei N die Anzahl der gesamten Usability-Probleme ist. L (in der Praxis $L=0.31$) ist das Verhältnis der Usability-Probleme, die eine Person findet, zu den gesamten Usability-Problemen. Die Anzahl der Testpersonen entspricht n.

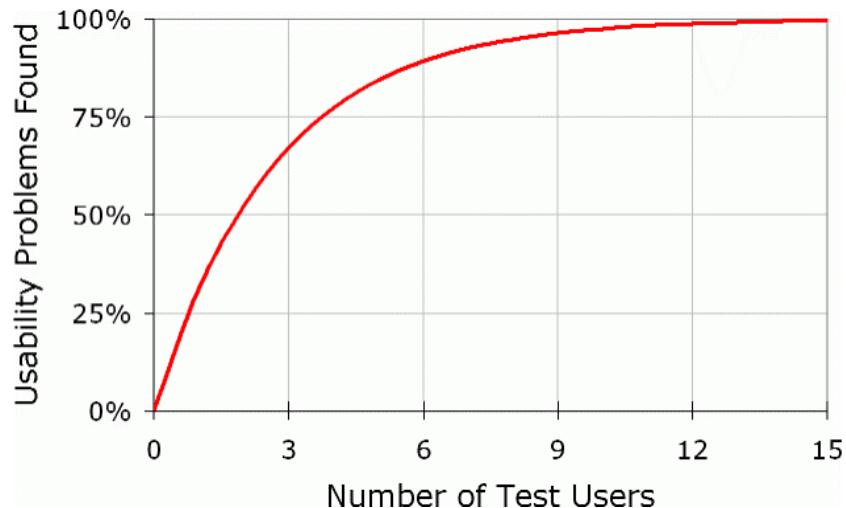


Abbildung 18: Gefundene Usability-Probleme [15]

2.5 Motivation für verwendete Methoden

Da diese Diplomarbeit auf die bereits vorhandene YPL.db aufbaut, werden auch die selben Methoden verwendet.

Diese sind im Einzelnen:

- Oracle 8 Datenbank
- Forms 6
- PL/SQL-TOOLKIT

3 Ergebnisse

In diesem Kapitel wird auf die Ergebnisse dieser Diplomarbeit eingegangen. Die Praktischen Arbeit hierfür wurde am Institut für Molekularbiologie, Biochemie und Mikrobiologie der Uni Graz ausgeführt.

3.1 Datenbankstruktur

Eine relationale Datenbank, die den Anforderungen in der Aufgabenstellung (siehe Punkt 1.3), gerecht wird wurde entworfen. YPL.db II ist in zwei Schemata aufgeteilt. Zum einen in das YPL-Schema, das die YPL.db II Datenbank enthält und zum anderen in das WYPL-Schema, das die PL/SQL-Prozeduren (siehe Punkt 2.2.2) für das Webinterface bereit stellt. Zum Schutz der Daten hat das WYPL-Schema nur Leserechte auf die fürs Internet freigegebenen YPL.db II Daten.

3.1.1 YPL Schema

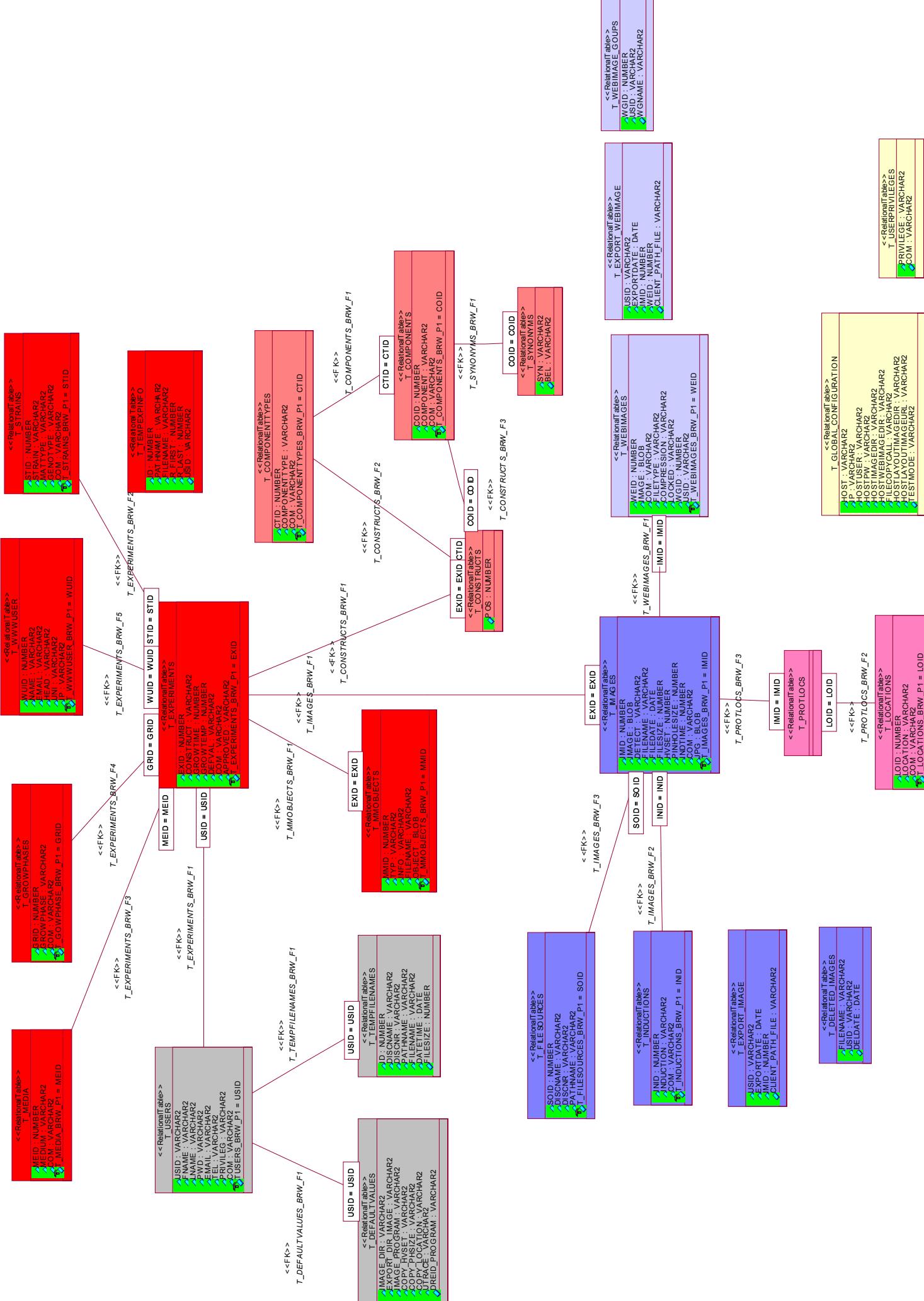


Abbildung 19: YPL Schema

In dem YPL-Schema sind die benötigten Tabellen für YPL.db II enthalten. Den Kern bildet hierbei die Experimenttabelle. Sie verweist auf die roten Experimentparameter Tabellen wie „growth phase“, „medium“, „growth temperature“. Die hellroten Tabellen beinhalten die Informationen über den genetischen Hintergrund des Konstruktes wie „plasmid“, „markers“, „promoter“ , und bilden somit die Quelle für die grafische Darstellung der GFP-Fusion (Siehe Punkt 1.1.5.3). Die Bilder und deren Parameter sind, in den blauen Tabellen, einem spezifischen Experiment zugeordnet. Die über das Web publizierten Bilder sind in den hellblauen Tabellen gespeichert. Somit ist eine Trennung der öffentlichen und internen Daten gegeben. In den rosa Tabellen sind die Lokalisierungen der Gene abgelegt. Für die Benutzerverwaltung, Datensicherheit und für Benutzer spezifische Einstellungen sind die grauen und gelben Tabellen zuständig.

3.1.2 WYPL Schema



Abbildung 20: WYPL Schema

Im WYPL-Schema sind die Tabellen für die Experimentübertragung mittels Internet enthalten, sowie die Pakete, mit den dazugehörigen Prozeduren, die das Webinterface für YPL.db II dynamisch generieren. Folgende Pakete sind enthalten:

EMAIL_HANDLER	→ Email Verständigung
INFO	→ Online Hilfetexte
SUBMIT	→ Experiment Übertragungs-Interface
TEST	→ Formular Überprüfung
WWW_STYLE	→ Style Konfiguration
YPL	→ YPL.db II Interface

3.2 Forms-Interface

In diesem Kapitel wird das Interface der YPL.db II beschrieben, welches mittels Forms Builder (siehe Punkt 2.3.1.1) erstellt wurde. Dieses Interface ist nur registrierten internen Benutzern zugänglich.

3.2.1 Login-Maske

Der Zugang auf YPL.db II erfolgt durch eine Login-Maske (siehe Abbildung 21: YPL.db II Login Maske), in der die User Id (erster Buchstabe vom Vornamen und die ersten beiden Buchstaben vom Nachnamen) und das zugehörige Passwort eingegeben wird.



Abbildung 21: YPL.db II Login Maske

3.2.1.1 Benutzerverwaltung

Verfügt der Benutzer über ausreichende Privilegien, so wird nach der Passworteingabe der Druckknopf „User Administration“ eingeblendet (siehe Abbildung 22: Benutzerverwaltung aufrufen). Damit kann die Benutzerverwaltung aufgerufen werden (siehe Abbildung 23: Benutzerverwaltungstabelle). Die Passwörter werden aus Sicherheitsgründen nur verschlüsselt dargestellt.



Abbildung 22: Benutzerverwaltung aufrufen

User								
	First Name	Last Name	UserId	Password	Password encoded	E Mail	Telephone	Privileg
Iskandar	Dib		IDI	xxxx	1V1V@@@@			,webimg,expimg,lovpre;
Georg	Habeler		GHA	x	□/+□@@@□@habeler@sbox.tu-graz.ac.at			;admin;alexp;lovpre,webimg,expimg,lovpre;
Mathias	Kals		MKA	x	□/+□@@@□@kalsm@sbox.tugraz.at			;admin;alexp;lovpre,webimg,expimg,lovpre;
Sepp	Kohlwein		SKO	x	□/+□@@@□@sepp.kohlwein@uni-graz.at			;admin;alexp;lovpre,webimg,expimg,lovpre;
Klaus	Natter		KNA	xxxxxx	□/3YOLY□@klaus.natter@uni-graz.at			;admin;alexp;lovpre,webimg,expimg,lovpre;
Saccharomyces	cerevisiae		SCE	xxxx	□/;VH□@@@□			;admin;alexp;lovpre,webimg,expimg,lovpre;
www_experimente	none		WWW	xxxx	□/?nn□@@@□@klaus.natter@uni-graz.at			;admin;alexp;lovpre,webimg,expimg,lovpre;

Abbildung 23: Benutzerverwaltungstabelle

3.2.1.2 Passwort setzen

In der Login-Maske kann das Passwort vom jeweiligen Benutzer neu gesetzt bzw. geändert werden (siehe Abbildung 24: Passwort ändern). Das neue Passwort muss zweimal hintereinander eingetragen werden, um Tippfehler auszuschließen.

Logon

First Name Mathias

Last Name Kals

User Id MKA

Password *

Logon Cancel Change Password

Abbildung 24: Passwort ändern

3.2.2 Haupt-Maske



Abbildung 25: YPL.db II Hauptmaske

Um die Übersichtlichkeit zu gewährleisten, ist die Hauptmaske in 7 Bereiche unterteilt.

3.2.2.1 Menüleistenbereich

Die Menüleiste bietet folgende Funktionalitäten an:

- Navigieren durch die Datensätze eines Blockes.
- Erstellen oder löschen eines Datensatzes in dem aktuellen Block (inklusive Detail-Datensätze).
- Durchsuchen der Datensätze eines Blockes.
- Speichern der ungespeicherten Änderungen.
- Beenden des Programms.



Abbildung 26: Menü Bereich

3.2.2.2 Konstruktbereich

Das Konstrukt des Experimentes wird hier angezeigt bzw. eingegeben. Das Konstrukt enthält Informationen über GFP-Variante, Promoter, Marker Gene und Konstrukttyp (integral, episomal oder vital stain). Die Elemente des Konstruktes werden mittels LOV (List Of Values, wird im LOV-Preset-Menü verwaltet) eingegeben (siehe Punkt 3.2.4). Dies geschieht durch Anklicken des benötigten Elementes und anschließender Auswahl aus der LOV.

Zur Beschleunigung der Eingabe, können im Konfigurations-Menü Konstruktvorlagen erstellt werden (siehe Punkt 3.2.4.1), die durch die Druckknöpfe „up“ und „down“ im Konstruktbereich ausgewählt werden können.

Nach der Eingabe des Elementes „Gene Name“ wird automatisch im Experimentbereich das Feld „Gene“ mit dem selben Namen gefüllt, um Fehleingaben zu vermeiden.

Abbildung 27: Konstrukt-Bereich

3.2.2.3 Experimentdatenbereich

Im Experimentdatenbereich werden die Daten zu jedem Experiment eingetragen bzw. angezeigt. Es sind hier Datenfelder vorhanden, die durch Anklicken über eine LOV befüllt werden. Diese Felder sind am Spaltenende mit einem kleinen LOV Schriftzug gekennzeichnet. Im „Config Menü“ (siehe Punkt 3.2.4.1) können „Default Experiment Werte“ gesetzt werden, die bei der Erstellung eines neuen Eintrages verwendet werden.

Besitzt der angemeldete Benutzer ausreichende Privilegien, so kann der Benutzer nicht nur auf die eigenen Experimente zugreifen, sondern auch auf deren aller vorhandenen Benutzer. Dies geschieht durch Umschaltung des „Radiobuttons“ auf „Select Experiments f. all Users“.

Das Importieren von Webexperimenten, die per Internet eingegeben wurden, ist auch von Privilegien abhängig und kann von berechtigten Benutzern mit dem Druckknopf „Import web experiment(s)“ aufgerufen werden.

Da für die Darstellung der Datensätze nur eine beschränkte Breite zur Verfügung steht werden Tooltips auf den Elementen verwendet, dadurch können auch überlange Einträge ohne scrollen angezeigt werden.

Es kann hier jede Spalte alphabethisch durch Anklicken des Spaltenkopfes sortiert werden.

Abbildung 28: Experimentdaten-Bereich

3.2.2.4 Bilddatenbereich

Hier werden die Detailinformationen über die Bilddaten eingegeben bzw. angezeigt. Es sind hier Datenfelder vorhanden, die durch Anklicken über eine LOV befüllt werden. Diese Felder sind am Spaltenende mit einem kleinen LOV Schriftzug gekennzeichnet. Im „Config Menü“ können hierfür „Default Experiment Values“ gesetzt werden, die bei der Erstellung eines neuen Eintrages verwendet werden. Im Bilddatenbereich können auch Webimages (Bilder die durch das Webinterface öffentlich zugänglich gemacht werden) erzeugt, bearbeitet oder gelöscht werden. Überdies können Webimage-Gruppen erstellt bzw. verwaltet werden, mit deren Hilfe ganze Sets von Webimages zusammengestellt bzw. freigegeben werden können.

Im Bilddatenbereich wird die Umschaltung des Experiment-Importmodes vorgenommen. Über die „Radiobuttons“ kann zwischen den Moden „Experimentlist“ (die Experimente werden in der Experimentliste aufgelistet und können Experimentweise ausgewählt werden), „Imagelist“ (die Bilder werden in der Bilderliste angezeigt und können bilderweise selektiert werden) und „Direct Load“ (die Bilder werden einzeln aus einem Verzeichnis geladen) umgeschaltet werden. Des Weiteren können durch Aktivieren des „Include Subdirectories“-Kontrollkästchens alle Unterverzeichnisse, die im ausgewählten Wurzelverzeichnis sind mit einbezogen werden.

Durch Drücken des „Export all Images“-Knopfes kann der gesamte Bilderstapel aus der Datenbank in ein lokales Verzeichnis exportiert werden. Dies dient zur weiteren Verarbeitung wie z.B. Erstellen von 3D-Bildern.

Auch hier werden Tooltips auf den Elementen verwendet um überlange Einträge ohne Scrollen anzeigen zu können.

Es kann hier auch jede Spalte alphabethisch durch Anklicken des Spaltenkopfes sortiert werden.

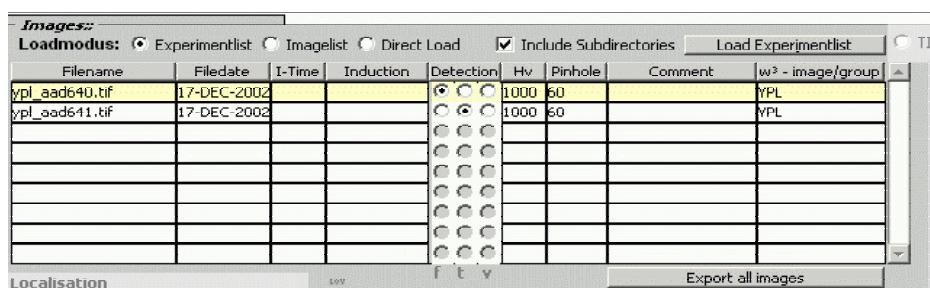


Abbildung 29: Bilddaten-Bereich

3.2.2.5 Lokalisierungsdatenbereich

Hier wird die Lokalisierung des Proteins angezeigt bzw. eingegeben. Die Lokalisierung kann nur über eine LOV eingetragen werden.

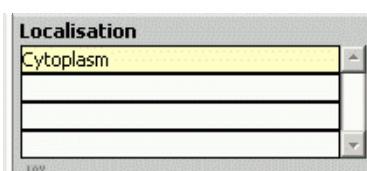


Abbildung 30: Lokalisierungsdaten-Bereich

3.2.2.6 Bildanzeigebereich

Hier werden die Bilder von dem Experiment angezeigt, die in der Datenbank gespeichert sind. Die Bilder können gegebenenfalls auch nachbearbeitet werden. Für die Nachbearbeitung muss im Konfigurations-Fenster (siehe Punkt 3.2.4.1) der Pfad eines Bildbearbeitungsprogramms eingegeben werden. Die Experimentbilder werden im TIFF- und im JPG-Modus in einem Rahmen mit der Größe von 256x256 Pixel angezeigt um Platz einzusparen. Um die Details eines Bildes erkennen zu können, kann mit den „+“ und „-“ das Bild skaliert werden und mit den Bildlaufleisten über das Bild gescrollt werden.

Mit den „Radiobuttons“ kann zwischen folgenden Modi gewählt werden:

- **TIFF:** Die Bilder, die in die Datenbank geladen werden, haben in der Regel eine Auflösung von 512x512 Pixel, eine Farbtiefe von 8Bit und liegen im TIFF-Format vor. Daraus resultiert ein Speicherbedarf von ca. 250kByte. Je nach Netzwerk und Netzauslastung kommt es zu zeitlichen Verzögerungen. Deshalb liegen die Bilder ebenfalls im JPG-Format vor, welche einen geringeren Speicherbedarf haben und folglich auch schneller geladen und angezeigt werden können. Aus diesem Grunde ist der JPG-Modus der Default-Modus. Der TIFF-Modus dient in erster Linie zur Darstellung der Originaldaten und deren Nachbearbeitung.
- **JPG:** Default-Modus für schnelleres Laden der Bilder. (siehe TIFF-Modus).
- **Image Gallery:** Um eine Übersicht über die vorhanden Bilder eines Experimentes zu bekommen, werden die Bilder in Form einer Gallery präsentiert. Durch Anklicken eines Bildes wird dieser Datensatz selektiert und die Gallery wird geschlossen.
- **Webimage:** In diesem Modus werden die vorhandenen Webimages angezeigt und können gegebenenfalls editiert werden.
- **3D:** Hier können Files von 3D-Bildern an ein Experiment gehängt oder gelöscht werden. Die 3D-Bilder werden im Webinterface als Link angeboten und können mittels VRML-Plugin im Browser dargestellt werden.
- **No Image:** Es werden keine Bilder geladen um schnellstmöglich durch die Datensätze navigieren zu können.

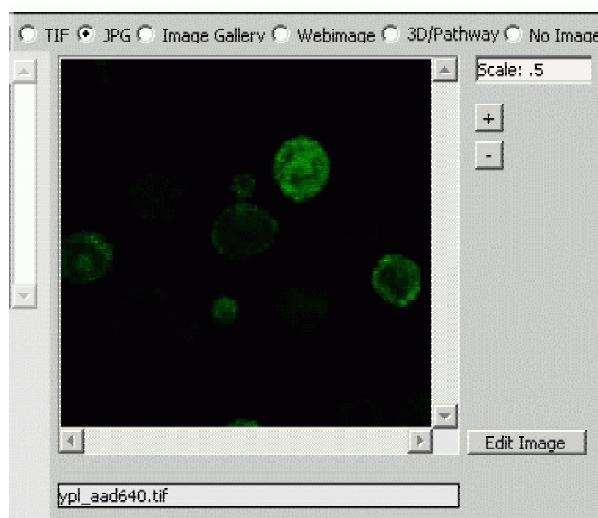


Abbildung 31: Bildanzeige Bereich

3.2.2.7 Listen-Bereich

Abhängig vom aktivierte Bildlademodus (siehe Punkt 3.2.2.4), wird hier die Experimentliste oder die Bilderliste angezeigt. Durch Doppelklicken auf ein Listenelement kann dieses aus dem lokalen Filesystem in die Datenbank importiert werden (Abbildung 25: YPL.db II Hauptmaske).

3.2.3 Webexperiment Import Masken

Die Webexperimente, die über das Webinterface (siehe Punkt 3.3.2) eingegeben wurden und noch nicht in YPL.db II übertragen wurden, sind hier aufgelistet. Sie können gegebenenfalls bearbeitet werden. Die Bilder der Experimente werden erst nach dem Drücken des „Show images“-Knopfes geladen und angezeigt. Durch Anklicken eines solchen Bildes wird es in voller Größe angezeigt (siehe Abbildung 33: WWW Import Image Fenster) und kann bei Bedarf nachbearbeitet werden.

Auch hier werden Tooltips auf den Elementen verwendet um überlange Einträge ohne Scrollen anzeigen zu können.

Es kann hier jede Spalte alphabethisch durch Anklicken des Spaltenkopfes sortiert werden.

The screenshot shows the 'WWWEXP' application window. At the top, there is a table titled 'Experimente:' with columns: Gene, USID, Strain, Genotype, Mating, Construct, Approved, Time, Temp, Medium, Growth phase, Induction, I-Time, and Localization. The first row contains data: TEST, MKA, CH1305, his3/his3; leu2/leu2; MET1a, sdfsfdsdf, Control PC, 12, 11, MM-his, early log, methionine, and lipid droplets. Below the table is a section titled 'Images:' which is divided into 'trans. images:' and 'fluor. images:'. Each section contains four image thumbnails, each with a 'Name:', 'Size:', and 'Date:' input field below it. At the bottom of the window are buttons for 'show image(s)', 'Drop current record', 'Submit current record', and 'Exit'.

Abbildung 32: Webexperiment-Import Übersicht

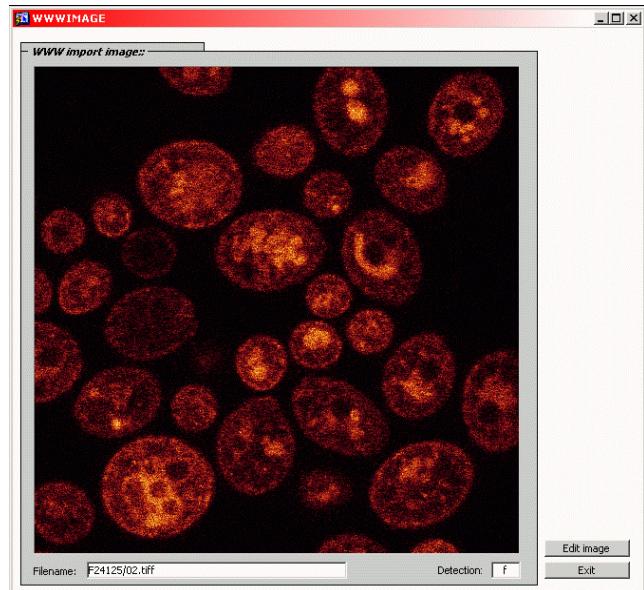


Abbildung 33: WWW Import Image Fenster

Nach dem Drücken des „Submit current record“-Knopfes öffnet sich das Add-Webexperiment-Fenster, welches die Objekte die automatisch neu erstellt werden in rot anzeigen. Hier muss auch das Konstrukt anhand des „Construct“-Feldes eingegeben werden (siehe Punkt 3.2.2.2).

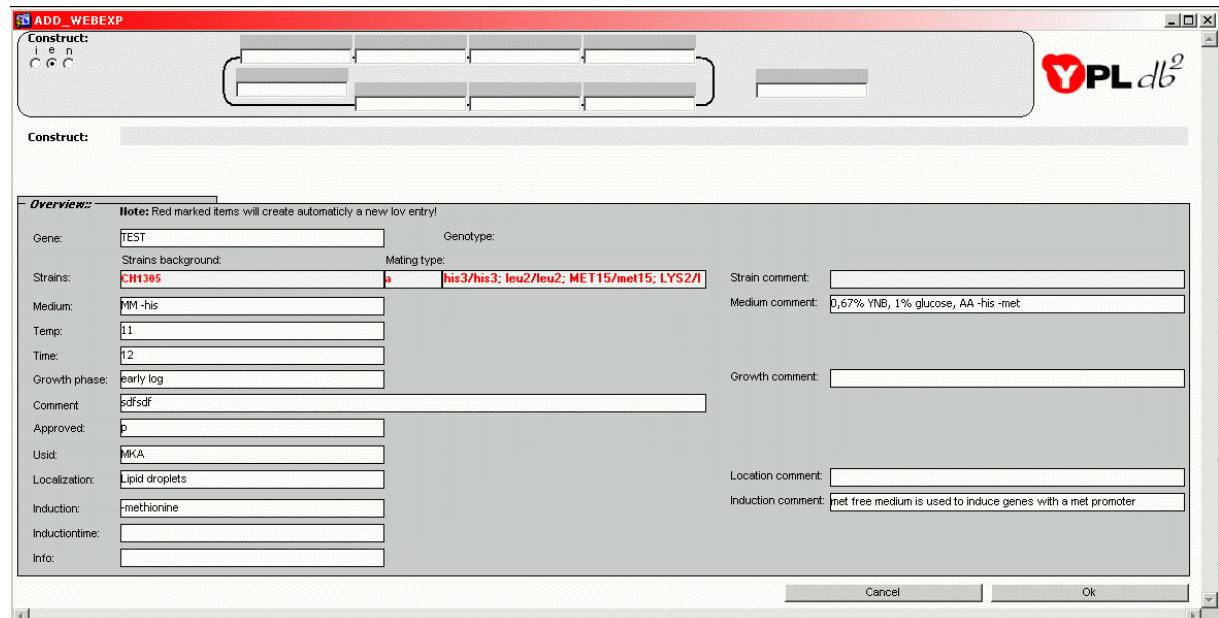


Abbildung 34: Web-Experiment-Import Fenster

3.2.4 Stammdaten-Maske

Benutzer mit ausreichenden Privilegien können diese Maske aus der Hauptmaske heraus aufrufen (siehe Punkt 3.2.2). Die Stammdaten werden hier verwaltet bzw. eingegeben. Die zur Erstellung der LOV's benötigt werden, die unter anderem die Lokalisation, Wachstumsmedien, Zellstämme usw. beinhalten.

Abbildung 35: Stammdaten-Maske

3.2.4.1 Konfigurations-Maske

Hier können benutzerspezifische Konfigurationen erstellt werden. Jeder Benutzer kann seine eigenen Default-Werte für Experimente setzen.

Main Form

Promoter TEF1	Gene Name AAC1	GFP Allele eGFP	
Backbone p416TEF			

YPL db²

Default Experiment Values:

Strain Background	Matatype	Genotype	Medium	Grow time	Grow temp	Growth Phase	Comment	Construct
FY1679	<input checked="" type="radio"/> a	ura3/ura3, his3, leu2, trp1	MM-ura plates	5	30	not defined		
CH1305	<input type="radio"/> a		YPD		30			
W303	<input type="radio"/> a	ura3/ura3, leu2, ade2, ade3	MM-ura -met		30			
								
								
								
								
								
								
								
								
								
								
								
								
								
								
								
								
								

Default Image Values:

I-Time	Induction	Detection	Hv	Pinhole	
		<input checked="" type="radio"/> f	<input type="radio"/> t	<input type="radio"/> x	Set Size
		<input checked="" type="radio"/> 700	<input type="radio"/> 50		Comment

Configurations:

Case Sensitive Query

set the same HVSET to all images from an experiment

set the same PINHOLESIZE to all images from an experiment

set the same LOCALIZATION to all images from an experiment

Image Export Directory:

Image Manipulation Program:

Abbildung 36: Konfigurations-Maske

3.3 Web-Interface

Für den öffentlichen Zugang wurden zwei Webinterfaces entwickelt. Eines für die Datenabfrage und deren Visualisierung und das andere zur Dateneingabe. Die Webinterfaces wurden mittels PL/SQL-Toolkit (siehe Punkt 2.2.2) erstellt. Um unterschiedliche oder falsche Darstellungen der Webseiten bei unterschiedlichen Browsern zu vermeiden wurden die W3C HTML 4.01 Guidelines angewendet. [22]

3.3.1 YPL.db II Abfrage Seite

Die Beschreibung des Abfrage-Interfaces befindet sich im Anhang unter YPL.db II Manual (siehe Punkt 7.4).

YPL HOME Yeast Genetics and Molecular Biology Group UNI Graz Bioinformatics Group TU Graz

Search: Gene: Search Clear
Localization:

Promoter: MET25 GFP Allele: eGFP Gene Name: ALR1
Backbone: pUG34 Marker: HIS3

Result:

Gene	Growth time [h]	Medium
ALR1	6	MM -his

1 rows found. Result set 1/1

Image Navigation:
< >
1/2
Show Gallery
Set bg. Image
no image selected!
:: Fade mode ::
Objects:
[3D] get wrl plugin

Detail:

Gene:	ALR1
Construct:	Episomal n-terminal
Strain:	GA74D-ko2
Genotype:	his3/his3
Localization:	Plasma Membrane
Medium:	MM -his
Growth Phase:	early log
Growth Time:	6 h
Growth Temp.:	30 °C
Construct verification:	none
Comment:	
Synonym List:	ALR1_00522_YOL130W SGD MIPS YPD REF

Comments? Contact [Webmaster](#)

Abbildung 37: YPL.db II Abfrage Seite

3.3.2 YPL.db II Experiment Übertragungs-Seiten

Für die Datenübermittlung wurde ein eigenes Interface entwickelt.

3.3.2.1 Experiment-Übertragungs-Login

Zur Übermittlung von Daten müssen sich die Benutzer vorher anmelden. Hierfür stehen zwei Möglichkeiten zur Verfügung. Registrierte (in YPL.db II vorhandene) Benutzer können sich mit ihrer Benutzerkennung und Passwort einloggen. Die anderen Benutzer müssen ihren Namen, ihre Emailadresse, das Labor und gegebenenfalls die Universität eingeben.

SUBMIT LOGIN

If you are a internal user then select your userid and enter your password.
Otherwise enter your name, email, head of laboratory and your university and department in section external user.

INTERNAL USER LOGIN:

Username	Password
<input type="text"/>	<input type="password"/>
<input type="button" value="reset"/>	<input type="button" value="login"/>

EXTERNAL USER LOGIN:

Your name	Your emailaddress	Head of laboratory	University and department
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="reset"/>	<input type="button" value="login"/>		

NOTE: Your ip-address will be logged!

Abbildung 38: Experiment-Übertragungs-Login

3.3.2.2 Experiment-Übertragungs-Eingabe

Hier werden die Daten der Experimente eingetragen und die Bilder in die Datenbank geladen. Durch Anklicken der Feldnamen wird ein Hilfesfenster zum jeweiligen Feld angezeigt.

YPL Image submission

Image submission sheet for YPL.db, for questions mailto: naki@box.tu-graz.ac.at.
For additional informations about columns click on table headers. With (*) marked fields are required and one of the (-) marked field also!

YOUR EXPERIMENTS:

strain info															growth info			select			
gene of interest	strain background	genotype	mating type	construct	construct verification	growth time [h]	temperature [°C]	medium	growth phase	induction	induction time [h]	localization	Image comment	reference	Microscope	other relevant information					
															<input type="button" value="delete last experiment"/>	<input type="button" value="delete all experiments"/>	<input type="button" value="submit all experiments"/>				

NEW EXPERIMENT:

strain info	gene of interest	<input type="text"/> (*)
	strain background	<input type="text"/> (*)
	genotype	<input type="text"/> (*)
	mating type	<input type="text"/> (*)
	construct	<input type="text"/> (*)
growth info	construct verification	<input type="text"/> (*)
	growth time [h]	<input type="text"/> (*)
	temperature [°C]	<input type="text"/> (*)
	medium	<input type="text"/> (*)
	growth phase	<input type="text"/> (*)
select	induction	<input type="text"/> (*)
	induction time [h]	<input type="text"/>
	localization	<input type="text"/> (*)
	transmission image(s)	currently no transmission image(s) uploaded!
	fluorescence image(s)	currently no fluorescence image(s) uploaded!
	image comment	<input type="text"/>
	reference	<input type="text"/>
	microscope	<input type="text"/>
	other relevant information	<input type="text"/>

Abbildung 39: Experiment-Übertragungs-Eingabe

3.4 Usabilitytest

Die Usabilitytests wurden von je 3 Biochemikern und Bioinformatikern durchgeführt. Diese Tests beschränkten sich lediglich auf das öffentlich zugängliche Webinterface (siehe Punkt 3.3) von YPL.db II, da dieses Webinterface besonders wichtig ist, um eine hohe Akzeptanz in der Yeast-Community zu erreichen. Bei den nur intern zugänglichen Forminterfaces (siehe Punkt 3.2) wird mehr Wert auf die Funktionalität gelegt, da es sich hierbei um speziell geschulte Benutzer handelt.

Zur Analyse der Usability wurde ein Testformular erstellt, welches Aufgaben für die Testpersonen enthält. Dieses Formular ist unter Punkt 7.6 zu finden.

Das Testformular enthält folgende Punkte:

- Anmelden am Webinterface.
- Ein Experiment per Internet übertragen.
- Nach einem bestimmten Gen suchen.
- Nach einer bestimmten Lokalisierung suchen.
- Durch Suchergebnis navigieren.
- Eine Bilder-Gallery von einem Experiment anzuzeigen.
- Ein Fluoreszenz-Bild mittels Fading über ein Transmissionsbild zu legen.
- Ein 3D-Bild anzeigen zu lassen.

3.4.1 Auswertung des Usabilitytests

Die Testpersonen mussten zu jeder Aufgabe, die im Testformular (siehe Punkt 6.6) vorgegeben ist, eine Bewertung der Kategorien Einfachheit, Vorhersehbarkeit, Transparenz, Effizienz und Fehlertoleranz angeben. Dies wurde mittels einer Zahl von 1 bis 5 vorgenommen. Wobei 1 für unzureichend steht und 5 für sehr gut. Von den Bewertungen der Testpersonen ist der Mittelwert (MW) für die Auswertung herangezogen worden. Des Weiteren wurden die Testpersonen angehalten, Bemerkungen einzutragen. Der genaue Testablauf ist aus dem Usabilitytest-Formular unter Punkt 7.6 zu entnehmen.

1. USABILITYTEST: ANMELDEN FÜR EXPERIMENTÜBERTRAGUNG					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	4	5	-	4
inf2	5	5	5	5	5
inf3	5	5	5	5	5
MW	5	4,83	5	5	4,83

Tabelle 8: 1. Usabilitytest: Login

Bemerkung der Testpersonen zu diesem Punkt:

- Fehlermeldung bei nicht korrektem Ausfüllen des Login-Formulars soll besser hervorgehoben werden.
- Feldinhalt sollte nach Fehlermeldung erhalten bleiben.

2. USABILITYTEST: EXPERIMENT ÜBERTRAGEN					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	3	5	-	4
inf2	5	5	3	3	5
inf3	5	5	4	5	4
MW	5	4,67	4,5	4,67	4,67

Tabelle 9: 2. Usabilitytest: Experiment übertragen

Bemerkungen der Testpersonen zu diesem Punkt:

- In Experimentübersicht sollten zu löschen Experimente selektierbar sein.
- Feedback beim Löschen und Speichern aller eingegebenen Experimente.
- Ein Exit Button zum Verlassen der Eingabeseite sollte vorhanden sein.

3. USABILITYTEST: GEN-SUCHE					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	5	5	5	5
inf2	5	5	5	5	5
inf3	5	5	5	5	5
MW	5	5	5	5	5

Tabelle 10: 3. Usabilitytest: Gen-Suche

Bemerkung der Testpersonen zu diesem Punkt:

- Wildcard-Suche sollte auch ohne Eingabe von ,* im Gene Feld funktionieren.

4. USABILITYTEST: LOKALISIERUNGS-SUCHE					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	5	5	5	5
inf2	5	5	5	5	4
inf3	5	5	5	5	5
MW	5	5	5	5	4,83

Tabelle 11: 4. Usabilitytest: Lokalisierungs-Suche

5. USABILITYTEST: DURCH SUCHERGEWINIS NAVIGIEREN					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	5	5	5	5
inf2	5	5	4	5	-
inf3	5	5	5	5	5
MW	5	5	4,83	5	5

Tabelle 12: 5. Usabilitytest: Durch Suchergebnis navigieren

Bemerkung der Testpersonen zu diesem Punkt:

- In der Resulttabelle sollte das aktuelle Gene farblich hinterlegt sein.

6. USABILITYTEST: BILDERGALLERY ERSTELLEN					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	5	5	5	5
inf2	5	5	5	5	5
inf3	5	5	5	5	5
MW	5	5	5	5	5

Tabelle 13: 6. Usabilitytest: Bildergallery erstellen

7. USABILITYTEST: BILDER FADING					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	2	3	3	3	5
inf1	5	5	5	5	5
inf2	5	5	5	5	5
inf3	5	4	5	5	5
MW	4,5	4,5	4,67	4,67	5

Tabelle 14: 7. Usabilitytest: Bilder Fading

Bemerkung der Testpersonen zu diesem Punkt:

- Die Opacity sollte auch manuell einstellbar sein.

8. USABILITYTEST: 3D-BILD ANZEIGEN					
Person	Einfachheit	Vorhersehbarkeit	Transparenz	Effizienz	Fehlertoleranz
chem1	5	5	5	5	5
chem2	5	5	5	5	5
chem3	5	5	5	5	5
inf1	5	5	5	5	5
inf2	5	5	5	5	-
inf3	5	5	5	5	5
MW	5	5	5	5	5

Tabelle 15: 8. Usabilitytest: 3D-Bild Anzeigen

Allgemeine Bemerkungen zu den Tests:

- YPL.db II Webinterface: Links zu den jeweiligen Bereichen (Search, Result usw.) auf das Online Usermanual wäre wünschenswert.
- YPL.db II Webinterface: Das Orange im Titelframe ist etwas zu grell für die weiße Schrift.
- Die Schriftgröße sollte generell größer sein, um eine bessere Lesbarkeit zu erreichen.

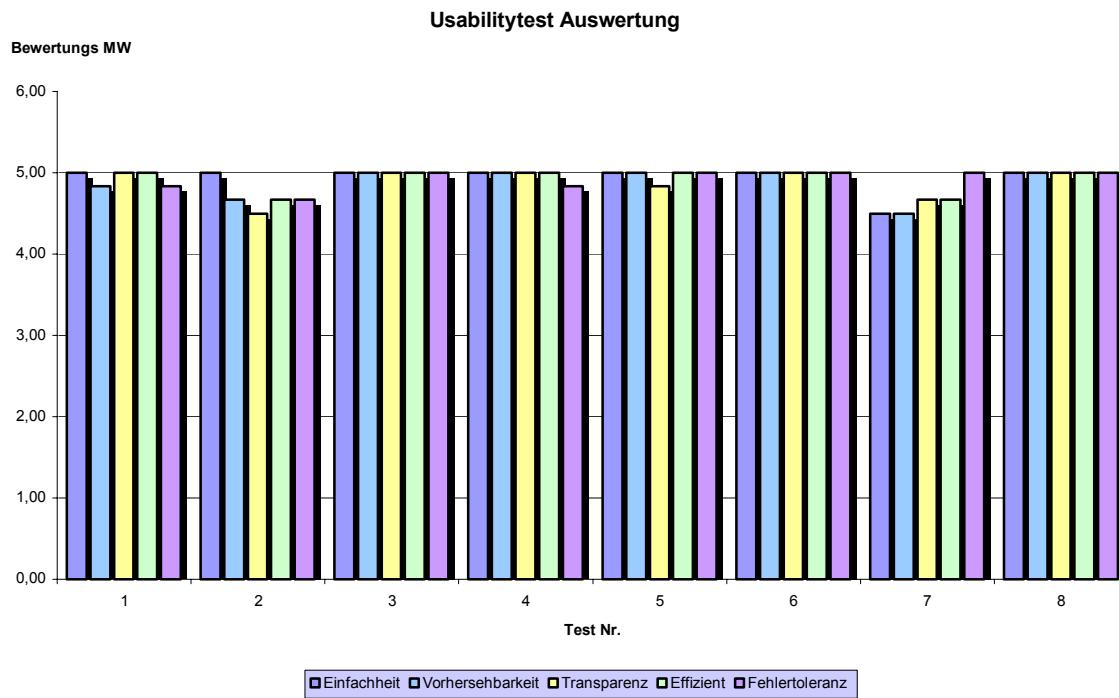


Abbildung 40: Graphische Auswertung des Usabilitytests

Wie die Abbildung 40: Graphische Auswertung des Usabilitytest zeigt hat das Webinterface bei dem Usabilitytest hohe Werte erreicht. Folglich ist auch eine gute Usability gegeben. Lediglich im Punkt 2 und Punkt 7 wurden etwas geringere Werte erreicht. Um die Usability zu steigen, sollten alle Bemerkungen der Testpersonen geprüft und gegebenen Falles auch umgesetzt werden.

4 Diskussion

YPL.db II stellt eine einfache webbasierte Möglichkeit dar, für die weltweite Yeast-Community, Fluoreszenz-Mikroskopie-Daten zu übermitteln und diese durch YPL.db II weltweit frei zugänglich zu machen.

Durch YPL.db II werden die Wissenschaftler bei der Auswertung der Fluoreszenz-Mikroskopie-Daten unterstützt. Es können Fluoreszenzbilder über Transmissionsbilder einer Zelle gelegt werden. Intrazelluläre Mechanismen werden dadurch besser veranschaulicht bzw. können leichter erkannt werden. Des Weiteren stehen für ausgesuchte Proteine 3D-Bilder zur Verfügung, welche aus den Mikroskop-Schnittbildern generiert wurden.

In dieser Arbeit wurden alle Anforderungen, die in der Aufgabenstellung gefordert sind, vollständig erfüllt. Großes Augenmerk wurde auf die gute Usability beim öffentlich zugänglichen Webinterface von YPL.db II gelegt, um eine größtmögliche Akzeptanz unter den Benutzern zu erreichen. Wie die Auswertung des Usabilitytests zeigt, wurde dieses Ziel erfüllt.

4.1 Sicherheit

Da sich YPL.db II immer noch im Aufbau befindet und fortlaufend neue Datensätze hinzukommen, ist es unbedingt notwendig eine zyklische Datensicherung der Datenbank vorzunehmen. Aus diesem Grunde wurde YPL.db II auf einem Server installiert, auf dem ein automatisches Backup läuft. Der eventuell mögliche Datenverlust wird dadurch gering gehalten und die Konsistenz der Daten ist sichergestellt.

Um die YPL.db II Daten vor unbefugtem Zugriff zu schützen ist der direkte Datenzugriff durch eine Benutzerverwaltung geschützt. Direkten Datenzugriff besitzen nur registrierte interne Benutzer mittels Forms-Masken (siehe Punkt 3.2).

Da YPL.db II im Internet unter <http://ypl.uni-graz.at> frei verfügbar ist, sind auch hier Sicherheitsaspekte für die Datensicherheit zu berücksichtigen. Oracle bietet mit dem OAS (siehe Punkt 2.2) ein sicheres Werkzeug an, um Datenbankinformationen einfach im Internet grafisch aufbereitet zugänglich zu machen. Selbst im Falle eines erfolgreichen Angriffes auf den OAS ist die Datensicherheit von YPL.db II gegeben, da OAS nur Leserechte auf freigegebene Daten von YPL.db II hat. Auch die Experimentdaten, die über das Internet übermittelt werden, können nicht direkt in YPL.db II übertragen werden, ohne dass diese durch einen Experten geprüft und freigegeben werden.

4.2 Perspektiven

Zur Zeit ist eine weitere Datenbank Yeast Lipid Metabolism Database (YLM.db) in Planung. In dieser soll YPL.db II ein integraler Bestandteil sein und Informationen über die Lokalisierung der Proteine bereitstellen.

Um die Browser-Abhängigkeit der Fading-Funktionalität zu vermeiden, sollte diese am Webserver in Perl mittels Imagemagic Library realisiert werden. Dadurch würde

YPL.db II Plattform und Browser unabhängig verfügbar sein. Die Anzeige des Fadings würde jedoch etwas langsamer werden, da bei einer Transparenz Änderung ein neues Bild generiert werden muss. Als Alternative könnte auch ein Java Applet für die Fadingfunktion implementiert werden.

5 Literaturverzeichnis

- [1] Brock T.D, Madigan M.T, Martinko J.M und Parker J; *Biology of Microorganisms* Seventh Edition; Prentice-Hall International Editions
- [2] Trachsel H; *Molekularbiologie*; <http://ntbionuser.unibe.ch/trachsel/teaching/mbvorl/mbvorl.htm#prokaryotisch>; Stand 23.07.2003
- [3] Löffler G; *Basiswissen Biochemie mit Pathobiochemie*; Springer
- [4] Schnupperkurs Biochemie; <http://www.oebvhpt.at/chemie/orville/index.html>; Stand 23.07.2003
- [5] Junqueira L.C, Carneiro J und Kelley R.O; *Histologie*; 5.Auflage; Springer
- [6] Bau der Zelle; <http://www.lambarene.de/pages/zellebau.html>; Stand 23.07.2003
- [7] Habeler G; Diplomarbeit YPL Bilddatenbank der Proteinlokalisierung in Hefezellen; Institut für Elektro- und Biomedizintechnische Technik; Technische Universität Graz
- [8] Stöcker H; *Taschenbuch der Physik*; Verlag Harri Deutsch
- [9] Confocal Microscopy; Leica Microscopy; <http://www.confocal-microscopy.com>; Stand 23.07.2003
- [10] Relationale Datenbanken; Fachhochschule Nordostniedersachsen
- [11] Grundlagen der Informationsmodellierung; <http://cis.cs.tu-berlin.de/lehre/ws-0203/sonstiges/gim-pages/pub/vl/152-188.kap3d.fm55.pdf>; Stand 23.07.2003
- [12] Lockman D, Debes N; Oracle 9i Datenbankentwicklung; *Markt + Technik*
- [13] Datenbanksysteme; <http://www3.informatik.tu-muenchen.de/lehre/ws2002/dbs-bayer/kap2-ermodelle.pdf>; Stand 23.07.2003
- [14] Odewahn A; *PL/SQL Developer's Indruction*; O'Reilly
- [15] Nielsen J und Landauer T; *Why You Only Need to Test With 5 Users*; <http://www.useit.com/alertbox/20000319.html>; Stand 20.07.2003
- [16] Meese E; http://www.uni-saarland.de/fak8/heinzle/de/teaching/bioinformatik_gpr_2003/2_mikroskopie_meese.pdf; Stand 20.07.2003
- [17] Codd E.F; *A Relational Model of Large Shared Data Banks*; Comm. ACM

- [18] Amira - Advanced 3D Visualization and Volume Modeling;
<http://amira.zib.de/>; Stand 20.07.2003
- [19] TOAD – Tool for Oracle Application Developers; <http://www.toadsoft.com/>;
Stand 23.07.2003
- [20] PL/SQL Developer; <http://www.allroundautomations.nl/plsqldev.html>; Stand
23.07.2003
- [21] Webalchemy; <http://www.users.bigpond.com/ahobbs/>; Stand 23.07.2003
- [22] World Wide Web Consortium; <http://www.w3.org/>; Stand 23.07.2003
- [23] Chen P; The Entity-Relationship Model: Toward a Unified View of Data.
ACM Transactions on Database Systems; M. Stonebraker
- [24] Usability; <http://www.gssi.de/leistungen/usability/presentation/>; Stand
04.09.2003

6 INDEX

#

β-Faltblatt	7
α-Helix	7

A

Absorption	
Absorption von Licht	11
Aminogruppe	6
Aminosäuren	2, 5
Amira	12
Anregungslicht	10, 11
Antikörperbildung	3
Atmungskette	1
Atomarität	24
Attribut	24
Attribute	18

B

Bäckerhefe	9
Bakterien	3
Benutzersichten	15
Benutzerverwaltung	37
Beugung	11
Beziehungen	23
Bilddaten-Bereich	41
Blaualgen	3

C

Carboxylgruppe	6
Codd	17
Construct	44
Cyanobakterien	3

D

DAD	26
Dateisystemen	15
Datenbank	14
Datenbank-Management-System	14
Datenbanksysteme	14, 15
Datenintegrität	23
Datenschutz	15

Datensicherung	15
Dauerhaftigkeit	24
DB	14
DB2	17
DBMS	14, 17
DDL	24, 25
Default-Werte	45
Detektor	10
Differenz	19
Direct Load	41
DML	24, 25
DNA	2, 3
dritte Normalform	24

E

Endoplasmatisches Reticulum	5
Energieniveau	11
Entity	24
Entity-Relationship Modell	25
Enzyme	7
ER-Modell	25
erste Normalform	24
Erythrozyten	1
Eukaryoten	2, 3, 4
Experimentdaten	40
Export all Images	41
exprimiert	9

F

Fluoreszenz	11
Fluoreszenzemission	11
Fluoreszenzen	10
Fluoreszenzlicht-Emission	11
Forms Builder	37
Fusionsprotein	8

G

GFP	8
Glattes endoplasmatisches Reticulum	1
Golgi-Apparat	1, 5
Green Fluorescent Protein	8

H

Hauptmaske	39
------------	----

Hefe	9	Webimage	42
HTML	27	Modellorganismus	9
HTTP-Listener	26	mySQL	17
<hr/>			
I		N	
Imagelist	41	Natürlicher Verbund	22
Informix	17	Normalisierungstheorie	23
Integration	14	Nukleinsäure	2
Interbase	17	Nukleoid	3
Interferenzen	11	Nukleus oder Nukleoid	2
Intrazelluläre Degradation	1	NULL	23
Isolation	24		
<hr/>			
K		O	
Katalog	15	OAS	26
Katalysator	7	operationale Integrität	24
Katalytische Proteine	7	Operationen	15
kloniert	9	Oracle	17
Kompartimentierung	4	Oracle Developer Suite	28
Konfig-Menü	40	Oracle Forms	28
Konfokale-Laser-Fluoreszenz- Mikroskopie	10	Organellen	1
konfokalen Mikroskop	10	oxidative Phosphorylierung	1
Konsistenz	24		
Konsistenzherhaltung	15		
Konstrukt	40		
Konzentrationsgradienten	3		
<hr/>			
L		P	
Lipide	2	Peptidbindung	5, 6
Login-Maske	37, 38	Peptide	5
Lokalisierung	41	Photomultiplier	10
LOV	40, 41	Photonen	11
Lysosome	1	PL/SQL	29, 30
<hr/>			
M		PL/SQL Developer	29
Makromoleküle	2	PL/SQL-Gateway	26
Membranproteine	3	PL/SQL-TOOLKIT	27
Menüleiste	39	Plasmamembran	3
Mitochondrien	1, 5	PostgreSQL	17
Mode		Posttranskriptionale Modifikation	1
3D	42	Primäre Protein Struktur	7
Image Gallery	42	Primärschlüssel	23
JPG	42	Privilegien	44
No Image	42	Produkt	18
TIFF	42	Projektion	21

Q	
Quartär Protein Struktur	8
Quotient	20
R	
Rational Rose	31
Raues endoplasmatisches Reticulum	1
RDBMS	23
referentielle Integrität	23
rekombinante DNA	8
Relation	18, 24
relationale Datenbank	34
relationale DBMS	14
Replication	1
RER	1
Ribosomen	1
Rückgrat	6
S	
Saccharomyces cerevisiae	9
Schnittbilder	10
Schnittmenge	20
Sekundäre Protein Struktur	7
Selektion	22
SER	1
show images	43
Spalte	24
SQL	25
SQL Server	17
SQL*Net	28
Stoffwechselreaktionen	4
Stokes-Verschiebung	10, 11
Struktur-Proteine	7
submit current record	44
subzelluläre Strukturen	1
Sybase	17
Synchronisation	15
T	
Tabelle	24
Tertiär Protein Struktur	8
TOAD	29
U	
Tooltips	40, 41, 43
Transaktion	24
Transaktionen	15
Transfektion	9
Tupel	18, 24
V	
Vakuolen	1
Vereinigung	19
Vesikeltransport bei Sekretionsprozessen	1
Viren	2
VRML	12, 13
W	
Wasserabspaltung	6
WebAlchemy	30
Webinterfaces	46
Wellenlänge	11
WRL-Plugin	42
WYPL-Schema	36
Y	
Yeast-Community	48
YPL-Schema	36
Z	
Zeile	24
Zelle	1
Zellkern	1, 3
Zellmembran	1, 3, 4
Zellorganellen	4
Zellplasma	3, 4
Zellwand	3, 4
zweite Normalform	24
Zytoplasma	2
Zytoplasma-Membran	2

7 Anhang

7.1 YPL Installations-Anleitung

Siehe YPL Installations CD Punkt Installmanual.

7.2 YPL Wartung & Änderungen

Bei Wartung bzw. Änderungen von YPL.db II ist mit CVS zu arbeiten. Das betroffene File wird zuerst ausgecheckt und nach der Bearbeitung wieder neu eingecheckt um eine neue Version zu erhalten.

7.2.1 Forms

Die Formsanwendung ist in vier Teilen unterteilt:

Adminmaske	→ YPLAdmin.fmb
Anmeldemaske	→ YPLLogon.fmb
LOVmaske	→ YPLLov.fmb
YPL Hauptmasken	→ YPLMain.fmb

Bei Änderungen müssen diese Files mittels Oracle Forms Builder 6i (siehe Punkt 2.3.1) bearbeitet werden und anschließend neu kompiliert werden. Dadurch entstehen Files mit der „fmx“ Erweiterung, die von der Forms Runtime (siehe Punkt 2.3.1) ausgeführt werden können.

7.2.2 PL/SQL

Die Webinterfaces wurden mittels PL/SQL-TOOLKIT (siehe Punkt 2.2.2) implementiert. Mehrere Pakete wurden hierfür unter dem USER „wypl“ erstellt. Diese erzeugen die dynamischen Seiten. Die Seite „ypl.htm“ z.B. wird durch das Paket „ypl“ und der darin enthaltenen Prozedur „htm“ generiert. Der Seitenname setzt sich also aus **<pakename>.<prozedurname>** zusammen.

Auf den Quellcode, der in der Datenbank liegt (Stored Procedures), kann mittels PLSQL-Developer (siehe Punkt 2.3.2), Toad (siehe Punkt 2.3.3) oder ähnlichen Werkzeugen zugegriffen werden. Auch hier ist mit CVS zu arbeiten, damit eine Versionskontrolle erfolgt.

7.3 Beschreibung (Projektstruktur, CVS)

Bei der Entwicklung von YPL.db II wurde CVS als Versionssystem benutzt. Folgende Projektstruktur wurde hierbei verwendet:

 - src	→ YPL.db II Quellcode
- plsql	→ PL/SQL Quell Files
- forms	→ Oracle Forms Files
 - documents	→ YPL.db II Dokumente
- htm	→ HTML Seiten
- pdf	→ PDF Seiten
- doc	→ Word Dokumente
 - tools	→ Programme für YPL.db II

Tabelle 16: YPL.db II Projektstruktur

7.4 YPL User Manual

Siehe auf YPL Installations CD unter Usermanual oder online auf http://www.oas.kfunigraz.ac.at:8010/ypl00/ypl_manual/default.htm.

7.5 YPL II Poster



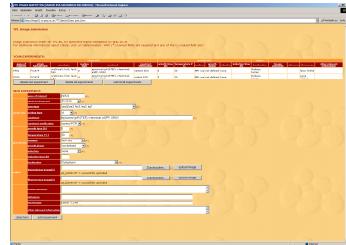
YPL.db - The Yeast Protein Localization Database

Mathias Kals (1), Klaus Natter (1), Heimo Wolinski (1), Georg Habeler (2), Gerhard Thallinger (2), Zlatko Trajanoski (2), **Sepp D. Kohlwein (1)**
 (1) IMBM-Biochemistry, University Graz, Schubertstr.1, Graz, A8010, Austria (sepp.kohlwein@uni-graz.at); (2) Institute of Medical Engineering, Technical University Graz, Austria



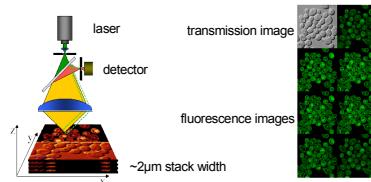
Abstract

Identification of protein function is a major task and challenge in the post genomic era. As part of the functional characterization we have started to address the spatial organization of proteins involved in lipid synthesis. Green fluorescent protein (GFP) provides a well-established tool to analyse the subcellular distribution of tagged proteins in living cells. Making use of a recombination cloning strategy, we have tagged some 550 yeast proteins, directly or indirectly involved in lipid synthesis, with GFP, and analysed them by confocal laser scanning microscopy. In order to document image data and other relevant information required for a correct interpretation of the localization patterns, YPL.db is the yeast protein localization database that contains data and parameters of the experiments together with images from confocal and video microscopy. The database can be queried based on gene name, protein localization or experimental parameters. YPL.db is an open platform that permits uploading of images from the yeast community, using a web-interface and mostly pre-defined lists for easy handling, and to ensure database integrity. The database also provides a structure reference resource to allow for better characterization of unknown or ambiguous localization patterns. Links to MIPS, SGD and YPD™ allow fast access to further relevant protein information not contained in the database. YPL.db is available at <http://YPL.uni-graz.at/>



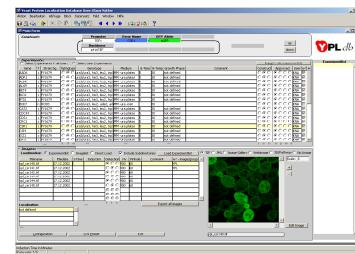
Web Based Image Submission

YPL.db is an image database open to all yeast researchers working on protein localization. Image submission is realized by means of a web accessible form; data entry is facilitated by online help. Integrity of submitted information is verified by an operator prior to publication in the database.



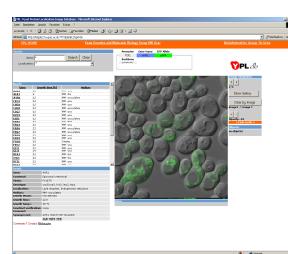
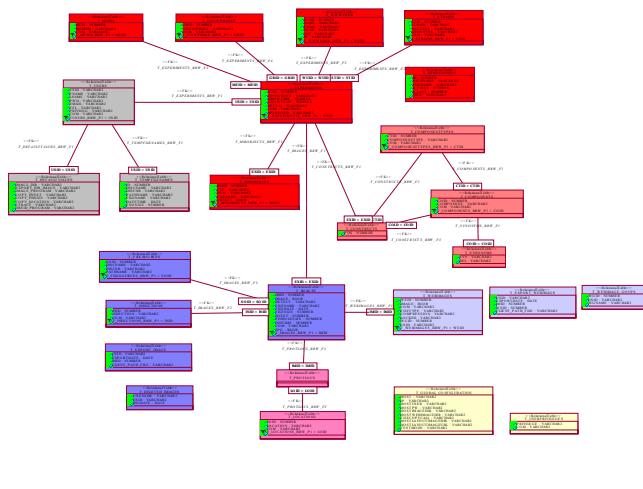
Confocal Laser Scanning Microscopy of yeast

Subcellular localization of GFP tagged yeast proteins is investigated at a large scale by confocal laser scanning microscopy. Either single optical sections or stacks of images for 3D visualization are recorded and stored for further processing.



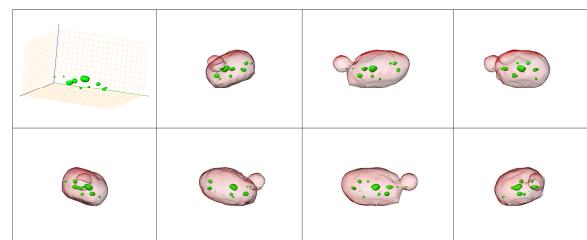
Data Entry and Administration of YPL.db

Oracle Forms is the interface for administration and storage of complete microscopy data. Basic functions besides editing experimental and strain data include image processing, integration of 3D images and fast query and sorting options. Additional features for web publication and review of data submitted through the web are implemented.



Publication of YPL Data

Public microscopy data is provided in a web accessible form together with information about the GFP construct and experimental conditions. Images are displayed in single or gallery modes. Fading allows overlay of fluorescence to transmission images in variable intensities. Implementation of 3D images and reference localization database are ongoing projects.



<http://YPL.uni-graz.at>



Acknowledgements

This work was supported by a grant from the Austrian Science Fund FWF, project F706 to S.D.K.



3D Image Visualization

Static 3D images are generated from stacks of single optical sections. Some data is further processed to animated graphics, presented with a virtual reality plug-in (VRML) on the web.

7.6 Usability Test Formular

Usabilitytest: Webinterfaces YPL.db II

Versuchsperson Nummer		Datum	
Versuchsperson Gruppe			

1. ANMELDEN FÜR DIE EXPERIMENTÜBERTRAGUNG

Die Seite <http://www.woas.kfunigraz.ac.at:8010/pls/al12/submit.logon> aufrufen. Felder im External User Login Bereich ausfüllen und Login drücken.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				

2. EXPERIMENT MITTELS WEBINTERFACE YPL.DB II ÜBERTRAGEN

Das Experiment ACB1 soll über das Webinterface von YPL.db II übertragen werden

Experimentdaten: Gene: ACB1; Strainbackground: FY1679; Matingtype: Diploid; Genotype: ura3/ura3, his3, leu2, trp1; Medium: MM -ura plates; Growth Time: 5; Growth Temp: 30; Growth Phase: not defined; Construct: Episomal, TEF1, C-Terminal, eGFP, URA3, p416TEF; Construct verification: none.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				

3. NACH GEN SUCHEN

Nach Experimenten für das Gene „ERG3“ suchen.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				

4. NACH LOKALISIERUNG SUCHEN

Nach Experimenten suchen welche die Lokalisierung „Cytoplasm“ besitzen.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				

5. DURCH SUCHERGEBNIS NAVIGIEREN

Anklicken eines im Result-Bereich gelisteten Experimententes und mittels drücken der „<“ und „>“-Knöpfe durch die Experimentbilder navigieren.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz

Bemerkung	
------------------	--

6. BILDER GALLERY ANZEIGEN

Durch anklicken des „Show Gallery“-Knopfes eine Gallery erstellen und ein Transmissionsbild durch anklicken auswählen.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				

7. BILDER FADING

Durch anklicken des „Set bg. Image“-Knopfes wird das Transmissionsbild gesetzt. Mittels „<“ und „>“-Knopf wird das zugehörige Fluoreszenzbild ausgewählt. Die Transparenz kann mittels „+“ und „-“-Knöpfe eingestellt werden.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				

8. 3D-BILD ANZEIGEN

Nach dem Experiment „ALR1“ suchen und im Objects-Bereich den Link „[3D]“ anklicken.

Einfachheit	Vorhersehbarkeit	Transparenz	Effizient	Fehlertoleranz
Bemerkung				