

# iSign – eLearning-System für die Mikrowellentechnik

Renate Wehrle,  
 Prof. Dr.-Ing. Andreas Christ  
 Fachhochschule Offenburg, Badstr. 24, 77652 Offenburg  
 Tel. 0781/205 - 0, Fax 0781/205 - 110

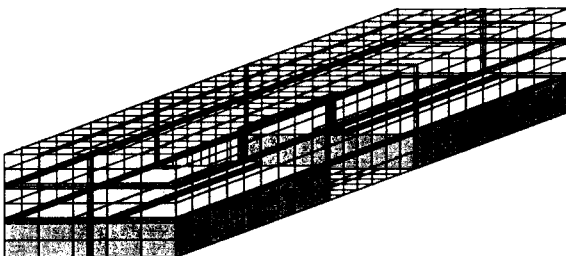
iSign — internet based simulation of guided wave propagation — ist eine Lernumgebung für Online-Laborversuche. Die Client-Serverarchitektur nutzt server-seitig das Tool F3D, das elektromagnetische Felder in 3D-Strukturen berechnet. Ein Apache-Webserver (unter Linux) bedient den Theorie-/Aufgaben-Teil und die Lernsystemadministration. Ein HPUX Simulationsserver steuert und kontrolliert den mehrstufigen Simulationsvorgang. Eine MySQL-Datenbank erlaubt dynamische Webseiten-Generierung und Simulations-, Projekt- und Userdatenhaltung. Java-Applets, JavaServer Pages und JavaBeans erzeugen die interaktive Client-Oberfläche zur Eingabe, Ergebnisdarstellung und für Online-Virtual Reality. Die einheitlich gestaltete Benutzeroberfläche verbirgt die Systemkomplexität.

## 1 Einführung

An der Fachhochschule Offenburg ist ein Softwaretool zur Simulation elektromagnetischer Vorgänge aus dem Bereich der Mikrowellentechnik im Einsatz. Das Softwaretool berechnet elektromagnetische Felder in dreidimensionalen, berandeten Strukturen. Hieraus ermittelt es die Ausbreitungskoeffizienten von Wellenleitern und die Streumatrix von Ein- oder Mehreren als zentrale beschreibende Größen des mikrowellentechnischen Verhaltens.

Abbildung 1 zeigt die Zerlegung der zu simulierenden Struktur in kleine, nicht äquidistante Quader. Diese dienen als Basis der zum Einsatz kommenden Finite-Differenzen-Methode zur Lösung des Maxwellischen Feldproblems.

Abbildung 1



3dimensionale Ansicht einer zu simulierenden Struktur

Das Softwaretool besteht aus FORTRAN-Programmen, die alle numerischen Berechnungen

vornehmen und die Ergebnisse in Dateien speichern. Sie enthalten die kompletten Berechnungsalgorithmen. Hilfsprogramme, die weitestgehend zeilenorientierte Eingaben verlangen, stellen die Benutzeroberfläche dar. Der Benutzer steuert den mehrstufigen Simulationsvorgang explizit durch das Starten der Programme. Die Auswertung der Ergebnisdateien erfolgte durch Übernahme der Ergebnisse in Standard-Grafikprogramme.

Das Forschungsprojekt iSign führt die einzelnen Komponenten zu einer integrierten Internetanwendung zusammen und integriert diese in eine umfangreiche Lernumgebung. Das eLearning-System beinhaltet sowohl die theoretischen Teile für das Selbststudium und zur Vorlesungsbegleitung als auch ein Simulations-Projektmanagement für selbständige und geführte Laborveranstaltungen.

Eine mehrstufige Client-Serverarchitektur mit zentraler Datenbank, dynamischer Webseitenerstellung, automatisierter Simulationsablaufsteuerung und einheitlich gestalteter Benutzeroberfläche bilden zusammen das eLearning-System iSign.

## 2 Anforderungen an das System

Ein Pflichtenheft charakterisiert die Anforderungen an das System. Inhaltlich, optisch wie auch funktional waren folgende Vorgaben einzuhalten.

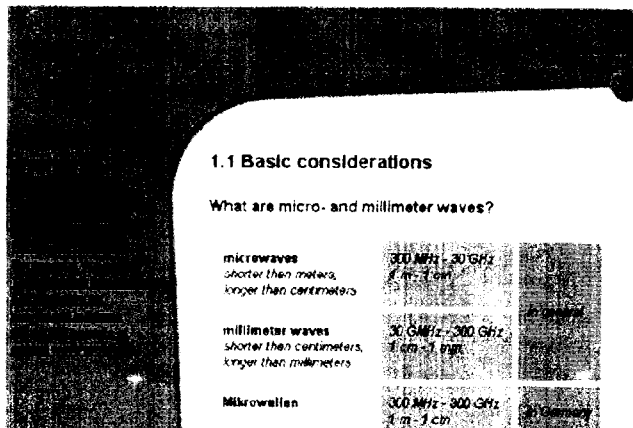
- **Benutzerfreundlichere Eingabe**  
Die Eingabe der Simulationswerte erfolgte bisher zeilenorientiert über eine Unix-Shell. Im eLearning-System erfolgt die Eingabe durch Eintragen von Werten in vordefinierte Formularfelder. Die Formulare werden sukzessive abgearbeitet, die Anwendung führt den Benutzer durch die Werteingabe. In späteren Versionen wird eine 3D-Modelling-Software in Java diesen Prozess unterstützen [2].
- **Integrierte Ergebnisausgabe**  
Zur grafischen Darstellung der Ergebnisse stehen Java-Applets zur Verfügung. Der Benutzer ruft diese aus der Lernumgebung heraus auf. Die interaktiv veränderbare Darstellung erfolgt innerhalb des eLearning-Systems.
- **Steuerung über das Internet**  
Der Zugriff auf die Simulation über das Internet

war in der bisherigen Anwendung nur mit netnet möglich. Moderne Internettechnologie und der Einsatz eines Datenbanksystems ermöglichen innerhalb des eLearning-Systems die Kontrolle des gesamten Steuerungsprozesses über das Internet.

- **Einbindung der Vorlesungsinhalte**  
Das Vorlesungsskript ist in das eLearning-System integriert. Dieses unterstützt die Vorlesungsveranstaltung und erlaubt das orts- und zeitunabhängige Selbststudium über das Internet.
- **Vertiefungsmöglichkeiten**  
Der Aufgabenteil und Auszüge aus Fachartikel vertiefen die Kenntnisse der Benutzer in der Thematik "Mikrowellentechnik".
- **Versuchsdurchführung**  
Individuelle Aufgabenstellungen, begleitende Theorie und die geführte Simulationssteuerung ermöglicht den Einsatz als Internet-Laborversuch.
- **Administration der Anwendung**  
Die Betreuung der Laborversuche und die Daten-, Benutzer- und Inhaltspflege der Anwendung ist über die Anwendung selbst gewährleistet.
- **Corporate Design**  
Das einheitliche Erscheinungsbild erleichtert die Orientierung innerhalb der Anwendung und verstärkt die Zusammengehörigkeit der einzelnen, inhaltlichen Elemente.

Abbildung 2 stellt die grafische Benutzeroberfläche der iSign Anwendung dar. Die Farbwahl richtete sich nach internen Vorgaben.

Abbildung 2



Farbliche Umsetzung der Benutzeroberfläche

### 3 Technologiewahl

Die Anforderungen des Systems können mehrere zur Auswahl stehende Technologien erfüllen. Da sehr viele unterschiedliche Komponenten benötigt werden, ist aber das Zusammenspiel der einzelnen Technologien untereinander maßgebend. Insbesondere die Servertechnologie ist für die Wahl der restlichen Komponenten entscheidend.

#### 3.1 Wahl des Servers

Innerhalb der Fachhochschule Offenburg kommt hauptsächlich der **Apache-Webserver** unter Linux zum Einsatz. Er ist einer der am weitesten verbreiteten Webserver weltweit. Seine Ausfallsicherheit, wie auch die Transparenz des offen gelegten Source-Codes, haben diese Vormachtstellung herbeigeführt. Als OpenSource Projekt wird der Source-Code ständig weiterentwickelt. Durch Zusatzmodule ist er für unterschiedliche server-seitige Technologien geeignet. Der Source-Code ist kostenlos über das Internet verfügbar.

#### 3.2 Wahl der Datenbank

Für den Apache-Webserver eignet sich das DBMS **mysql**. Dieses ebenfalls kostenlos verfügbare Datenbanksystem hat für Internetanwendungen eine sehr gute Performance. Die Ausfallsicherheit ist mit Oracle vergleichbar, die Wiederherstellung der Daten nach einem Ausfall sogar oftmals noch besser. Die Anforderungen an ein relationales DBMS sind zwar nicht konsequent umgesetzt, spielen aber im Internetbereich eine eher untergeordnete Rolle. Die "lamp" Konstellation (linux, apache, mysql, php) wird oft für kleinere eBusiness-Lösungen in der Industrie eingesetzt.

#### 3.3 Wahl der client-seitigen Technologie

Die visuelle Darstellung der Ergebnisse lokal beim Client übernehmen **Java-Applets**. Die Standardbibliotheken von Java enthalten diverse Zeichenobjekte und interaktive Steuerungselemente (Buttons, Listen, Auswahlboxen,...).

Die Kommunikation der verteilten Anwendung erfolgt über Sockets. Diese übernehmen den Transfer der Anwendungsdaten. Die Ergebniswerte des Simulationsdurchlaufes werden vom Webserver zum Client transferiert. Die Werte befinden sich client-seitig nur im Arbeitsspeicher.

Applets haben aus Sicherheitsgründen keinen Zugriff auf die Festplatte des Client-Rechners. Speziellen "gezeichneten" Applets kann der Benutzer explizit Schreib- und Leserechte einräumen. Im Falle von iSign ist aber keine Datenspeicherung beim Client vorgesehen, da die Datenmanipulation aufgrund des Einsatzes als Laborversuch nicht erwünscht ist.

#### 3.4 Wahl d. server-seitigen Technologie

Server-seitige Technologie gewährleistet die Generierung dynamischer Webseiten. Die Dynamik übernimmt Programm-Code, der die anwendungsbezogenen Daten verarbeitet. Die verfügbaren Techniken unterscheiden sich durch Aufenthaltsort und Sprache des Programmcodes. Die folgenden Techniken sind in der Internetprogrammierung zur Zeit verbreitet:

- **cgi-Programmierung**  
Ausführbare Programme in PERL oder C. Sie be-

finden sich auf dem Webserver und werden durch http-Anfragen (request), die der Benutzer über den Browser auslöst, aufgerufen und auf dem Server ausgeführt. Die Standardausgabe dieser Programme übergibt HTML-Code über die http-Antwort (response) an den Client. Dieser Code enthält dynamische Ausgaben, die HTML-Programmierung alleine nicht bewerkstelligen kann.

- **Server API's**  
Schnittstellen, die Protokolle zur Verfügung stellen. Die server-interne Struktur wird zur Verfügung gestellt. Durch umfangreiche Möglichkeiten entsteht ein hoher Einarbeitungsaufwand. Ein Beispiel für eine Server API wären Java-Servlets von Sun.
- **Server Side Includes**  
Hier fällt keine Programmierung im eigentlichen Sinn an. Die Erstellung der dynamischen Ausgabe übernehmen Tags, die der Server bereitstellt und von diesem ausgeführt werden. Der HTML-Code erhält eine spezielle Endung und enthält diese SSI-Tags.
- **server-seitige Skripte**  
Der Programmcode ist bei dieser Technologie in die HTML-Seiten eingebettet. Beim Aufruf der Seite, wird dieser Code, im folgenden als Skript bezeichnet, interpretiert. Erzeugt das Skript eine Ausgabe, wird diese an dessen Stelle eingefügt. Der Client erhält nur eine HTML-Seite. Da diese keinerlei Skripte mehr enthält, ist die ursprüngliche Zusammensetzung der Seite nicht mehr ersichtlich.
  - **PHP (personal home page)**  
Diese Technologie ist weit verbreitet und benutzt eine einfache Hypertext Präprozessor Skriptsprache. Ein Zusatzmodul für den Apache-Webserver ist verfügbar. Eine direkte Kommunikationsverbindung von PHP zu Java-Softwarekomponenten ist dagegen nicht möglich.
  - **ASP (active server pages)**  
Der Microsoft Internet Information Server enthält standardmäßig diese Erweiterung. Die verwendete Skriptsprache ist VisualBasicScript.
  - **JSP (java server pages)**  
Die verwendete Skriptsprache ist Java. Der Zugriff auf Java-Bibliotheken erleichtert die Anwendungsdatenverarbeitung. Durch die Installation des Zusatzmodules "Tomcat" kann der Apache-Webserver JSP verarbeiten.

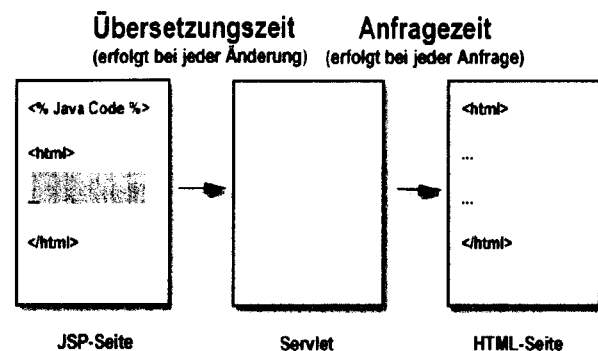
Diese Technologien ermöglichen den Zugriff auf Datenbanken oder andere Dienste (Mail, Datumsanzeige, ...). Sie unterscheiden sich durch ihre Funktionalität und durch ihre Funktionsweise.

Der Einsatz von **JSP** begründet sich damit, dass durch die vollwertige Programmiersprache wesentlich komplexere Möglichkeiten vorhanden sind.

Die Programmiersprache Java ist auch für die Netzwerkkommunikation und für die Darstellung der Ergebnisse im Einsatz. Eine homogene Java-Anwendung ohne eine zusätzliche server-seitige Skriptsprache entstand. Dadurch war es möglich Softwarekomponenten (JavaBeans), Codefragmente und Java Know-How auszutauschen.

Ein weiterer Aspekt für JSP-Technologie ist die Schnelligkeit. Eine JSP-Seite wird nur dann kompiliert, wenn sie verändert wurde. Eine JSP-Seite ist nach der Kompilierung ein Java-Servlet (Abb. 3). Dieses Servlet übernimmt die Erstellung der dynamischen Inhalte. Das bedeutet, der Entwickler unternimmt durch Testen der Seite diesen ersten Aufruf. Alle weiteren Anfragen von Benutzern gelangen zu den schon kompilierten Servlets.

Abbildung 3



Das kompilierte Servlet übernimmt die Ausgabe bei jeder Anfrage

## 4 Datenbankentwurf

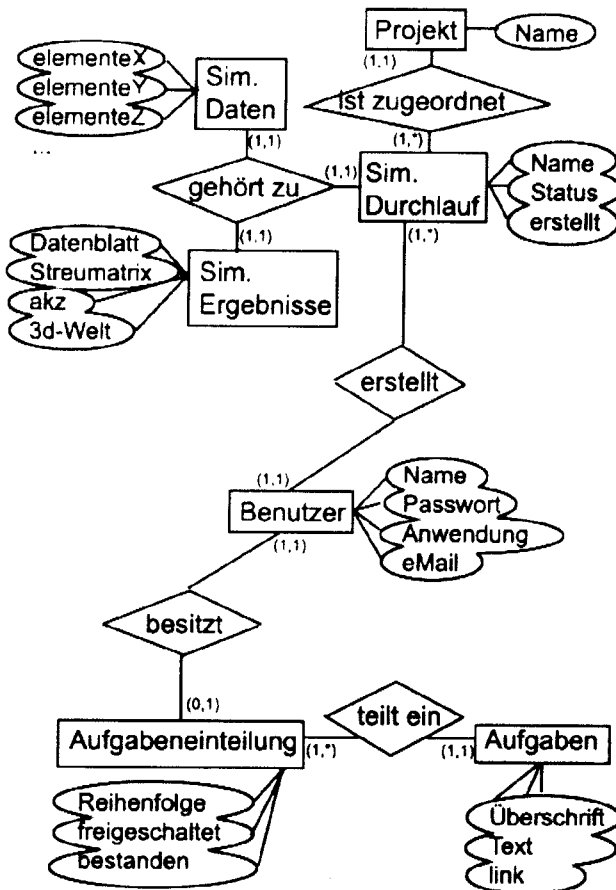
Das Zusammenspiel von Datenbank, JSP-Seiten und Webserver ermöglicht die Generierung dynamischer Inhalte. Die logische Strukturierung der Daten vermeidet Fehlfunktionen der Anwendung im laufenden Betrieb.

Ein wichtiges Ziel des Datenbankentwurfes ist die Abbildung realer Zustände in Daten des Datenbanksystems. Die genaue Beschreibung dieser Zustände erfolgte durch Ablaufdiagramme. Diese Szenarien legten somit den Ausschnitt der realen Welt fest. Die Klassifizierung von Objekten hilft bei der Erstellung von ER-Diagrammen. Die Einteilung der Elemente in eigenständige Objekte, Eigenschaften eines Objektes oder in Beziehungen verdeutlicht den Vorgang.

### 4.1 ER-Diagramm

ER-Diagramme und anschließende Normierung vermeidet Redundanzen im Datenbankentwurf. Wird Redundanz vermieden, entstehen durch Dienstfunktionen keine Inkonsistenzen im laufenden Einsatz. Das ER-Diagramm der Simulationsdurchführung im Rahmen eines Laborversuches skizziert Abbildung 4.

Abbildung 4



ER-Diagramm für Simulationssteuerung mit Laborversuch

Objekte/Relationen sind als Rechtecke dargestellt, Beziehungen zwischen Objekten stehen in Rauten und Attribute von Objekten sind von Ellipsen umgeben. Die (min, max)-Notation neben den Objekten legt fest, wie oft die Beziehung zwischen den Objekten minimal und maximal vorkommt.

## 4.2 Tabellenentwurf

Die ER-Diagramme dienen als Grundlage erster Tabellenentwürfe. Die Objekte wurden zu Relationen (Tabellen) und erhielten ein zusätzliches Schlüsselattribut (Identifikator, Primärschlüssel). Das Schlüsselattribut beschreibt das Objekt eindeutig. Beziehungen zwischen den Relationen beschreiben Fremdschlüssel. Diese sind Primärschlüssel in einer anderen Relation und stellen so die Verbindung zwischen diesen beiden Relationen her.

In Abbildung 5 wird die Tabelle "User" dargestellt. Diese ist für die Einteilung der Benutzer in Anwendergruppen (s. Kapitel 6.2) zuständig. Das Attribut user\_ID ist Primärschlüssel. Jeder Benutzer besitzt seine eigenen Simulationsdaten und muss daher in der Datenbank eindeutig identifizierbar sein.

Abbildung 5

user_ID	username	password	application
1	Christ	.....	superuser
2	et_1	.....	et
3	et_2	.....	et
4	cme_1	.....	cme
5	cme_2	.....	cme
6	Admin	.....	admin
15	x	.....	guest

Tabelle "User" für die Anwendung iSign

## 4.3 Normierung

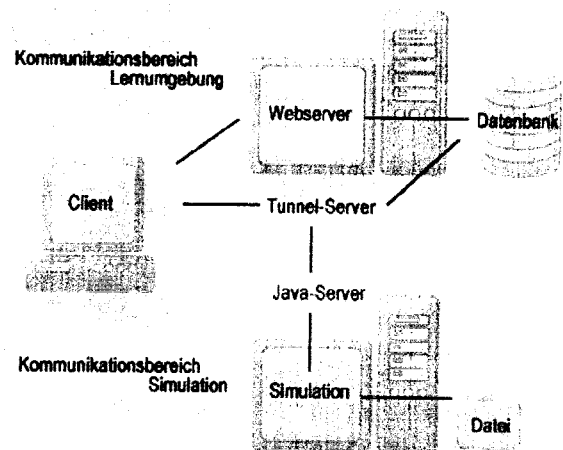
Die Normierung überprüft den Tabellenentwurf auf seine Güte. Die Normalform ist ein Maß für die Güte eines Entwurfes. Je höher die Normalform des Entwurfes, desto besser ist seine Güte. Ab der 3. Normalform spricht man von einer ausreichenden Güte des Entwurfes.

Die Motivation, die Tabellen nach diesen Gesichtspunkten zu bewerten und zu verändern, begründet sich aus der Vermeidung von *Anomalien*. Ansonsten könnten im praktischen Einsatz Dienstfunktionen die Datenbank in inkonsistente Zustände versetzen, wenn bei deren Durchführung eine Störung auftritt.

Die Tabellen des iSign-Entwurfes befinden sich alle mindestens in der 3. Normalform.

## 5 Systemarchitektur

Abbildung 6



Komponenten der iSign Anwendung

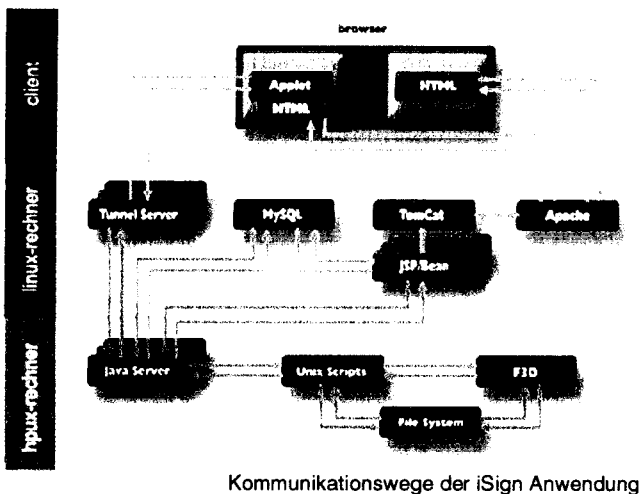
Die Client-Server-Anwendung ist auf mehrere Rechner verteilt. Der Webserver-Rechner, der Simulations-Rechner und der Rechner des Benutzers sind an der Anwendung beteiligt (Abb. 6). Auf jedem Rechner befinden sich mehrere Komponenten des Systems.



## 5.1 Komponenten des Systems

Die Abbildungen 6 und 7 zeigen den physikalischen Aufenthaltsort der einzelnen Komponenten. Auf dem Webserver-Rechner (Linux-Rechner) befinden sich die Apache-Software, die MySQL-Datenbank, Tomcat für den Einsatz der JSP-Technologie und eine Java-Applikation (Tunnel-Server) zur Netzwerkkommunikation. Der Simulations-Rechner (HPUX) verwaltet die Simulationsergebnisse, führt die Simulations-Programme aus und enthält auch einen Java-Server für die Netzwerkkommunikation.

Abbildung 7



## 5.2 Kommunikationswege des Systems

Die einzelnen Technologien tauschen untereinander Anwendungsdaten aus. Java-Applets benötigen z.B. Inhalte von Dateien, die sich auf dem Simulations-Rechner befinden, JavaBeans stellen eine Verbindung zwischen JSP-Seiten und der Datenbank her und der Webserver liefert HTML-Seiten an den Client. Abbildung 7 verdeutlicht die Verbindungen zwischen den einzelnen Komponenten. Klare Schnittstellendefinitionen vermeiden "Verständigungsprobleme" zwischen den einzelnen Komponenten.

Durch den Einsatz von JavaServer Pages konnte auch zu den Java-Servern eine direkte Verbindung aufgenommen werden. Eine Entscheidung für php3 oder php4 hätte dies ausgeschlossen.

## 6 Aufbau der Lernumgebung

Das eLearning-System kann in unterschiedlichem Lernkontext eingesetzt werden. Der Benutzer arbeitet und lernt innerhalb der an seine Lernanforderungen angepassten Umgebung. Die Zuweisung zu einer Anwendergruppe mit zugehörigen Nutzungsrechten übernimmt die vorgeschaltete Anmeldung (Abb. 8). Diese schützt auch vor unzulässigen Zugriffen auf das eLearning-System.

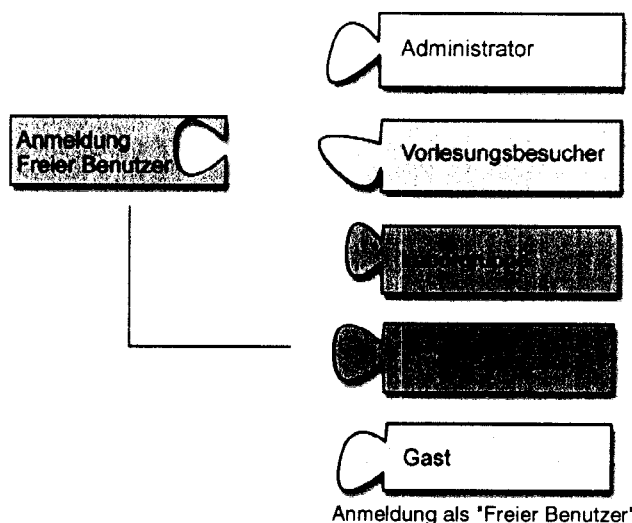
## 6.1 Anmeldung

Mit gültigem Benutzernamen und Passwort gelangt der Benutzer zu seiner Lernumgebung. Gültige Benutzernamen, die Zuordnung zur Anwendergruppe, die Vergabe von Lernaufgaben usw. werden vom Administrator verwaltet und den autorisierten Benutzern mitgeteilt.

Über einen Gastzugang ist ein freier Zugang offen. Jeder dem System unbekannte Benutzername wird als Gast eingestuft. Dieser kann eine eingeschränkte Version des eLearning-Systems nutzen.

Die Beschränkung des Umfangs erfolgt durch das "nicht-vorhanden-sein" von Manipulationsmöglichkeiten und Restriktionen im Simulationswerkzeug.

Abbildung 8



## 6.2 Anwendergruppen

Die vorhandenen Anwendergruppen gehen auf das unterschiedliche Nutzungsverhalten der einzelnen Benutzer ein. Die Inhalte der Anwendung variieren:

- Administration:  
Benutzerverwaltung, Versuchsverwaltung, Statistikabfrage, Seitenpflege, Datenpflege
- Vorlesungsbesucher:  
Theorie, Übungen, Fachartikel
- Laborgruppen:  
Theorie, Übungen, Fachartikel, Versuchsdurchführung
- Freie Simulation:  
Unbeschränkte Simulation, Projektverwaltung
- Gast:  
Theorie, eingeschränkte Simulation, Fachartikel

## 6.3 Inhalt

Die einzelnen Menüpunkte, die je nach Anwendergruppe in der Kopfzeile dargestellt werden (Abb. 2), enthalten folgenden Inhalt.

- Theorie  
Vorlesungsskript und/oder Beschreibung der Simulationssteuerung.

- Übungen  
Übungsaufgaben und Klausursammlung zur Vorbereitung auf die Klausuren.
- Fachartikel  
Auszüge aus Veröffentlichungen und Neuigkeiten innerhalb der Fachhochschule Offenburg zum Thema Mikrowellentechnik.
- Simulationsumgebung  
Fragenkatalog mit Verständnisfragen zur Simulation, Steuerung der Simulation, evtl. Aufgabenstellung des Laborversuches und Projektverwaltung.

Der Administrator erhält andere Menüpunkte. Die Anwendung für den Administrator wird zur Zeit entwickelt und steht somit noch nicht im Detail fest. Die Anforderungen des Pflichtenheftes sollen erfüllt werden.

#### 6.4 Analyse der Simulationsergebnisse

Zur Analyse der Ergebnisse werden Java-Applets eingesetzt. Diese ermöglichen die Darstellung der Ausbreitungskoeffizienten, der Streumatrix und der elektromagnetischen Felder [3].

Abbildung 9 zeigt das User Interface des Java-Applets zur Visualisierung der Streumatrix. Es verstärkt den Lernerfolg bei der Auswertung der Ergebnisse, da der Lernende sich primär auf die Ergebnisinterpretation konzentrieren kann.

Die Einbindung des Simulationswerkzeuges in die Lernumgebung gestattet die äußerst flexible Analyse realer Strukturen über das Internet. Ein physikalischer Versuchsaufbau mit über das Internet gesteuerten Messungen ist in seiner Variabilität sehr viel eingeschränkter. Gleichzeitig wird der Umgang mit Simulatoren trainiert. Es wird vermutet, dass der Lernerfolg deutlich höher ist.

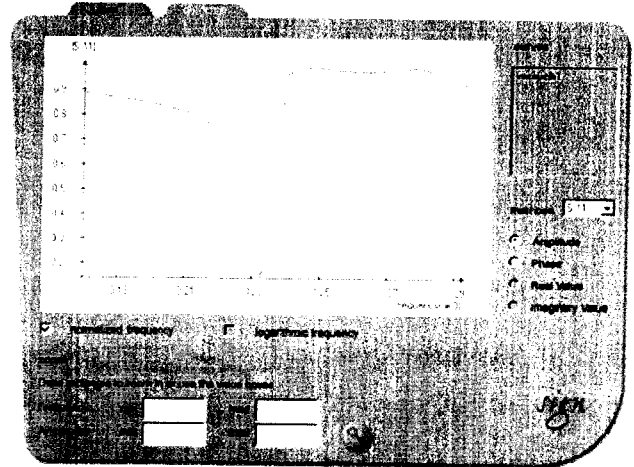
### 7 Zusammenfassung

Ungewöhnlich bei der vorgestellten Lernumgebung ist der Ansatz der praktischen Umsetzung. Es werden nicht nur Lerninhalte angeboten, sondern der Benutzer testet das Erlernete in der Simulationsumgebung und kann in dieser auch seine Ergebnisse auswerten. Die speziell zugeschnittene Software benutzt konsequent Internettechnologie um zusätzliche Mehrwerte zu schaffen.

Ein wichtiger Aspekt ist der Einsatz als Laborversuch [4]. Praktische Laborleistungen an Fachhochschulen sind bisher an eine Anwesenheitspflicht gekoppelt. Diese garantiert, dass jeder Student diese Leistung selbständig erbringt. Die Funktionsweise der Simulationssteuerung gewährleistet einen Kopierschutz und verhindert somit Manipulationen während der Durchführung des Versuches. Die Ergebnisse der Simulationen werden nicht lokal beim Benutzer gespeichert. Er kann die Werte nachträglich nicht verändern. Entsprechendes gilt für die Eingabewerte für die Simula-

tion. Diese werden in der Datenbank gespeichert. Die Einträge sind nur für autorisierte Benutzer zugänglich. Damit kann kein Dritter Eingabeparameter oder Ergebnisse kopieren und als seine ausgeben. Dieser Umstand ermöglicht der Simulationssteuerung den zusätzlichen Einsatz als Internet-Laborversuch.

Abbildung 9



Darstellung des Applets für die Streumatrix

### 8 Ausblick

Die Software findet mit Beginn des Wintersemesters 2001/02 erstmals ihren Einsatz als eLearning-Umgebung. Anhand des praktischen Einsatzes wird überprüft, in wie weit die gestellten Anforderungen an die Anwendung auch für studentische Benutzer vernünftig gestaltet sind.

Generell wäre der Einsatz von XML-Technologie empfehlenswert. Diese könnte die interne Automatisierung der Lernumgebung und die Ablaufsteuerung der Simulation verbessern. Die Speicherung von XML-Daten in Datenbanken ist zum jetzigen Zeitpunkt allerdings noch nicht ausgereift. Deshalb wurde dieser Implementierungsansatz vorerst zurückgestellt.

### Literatur

- [1] Möller, O.: Dokumentation: "iSign". Fachhochschule Offenburg, 2001.
- [2] Feißt, M.: Interactive 3D User Interface. Master Thesis Fachhochschule Offenburg, WS 2000/01.
- [3] Feißt, M.; Christ, A.: Virtual-Reality-Darstellung elektromagnetischer Felder in dreidimensionalen Mikrowellenstrukturen. Multiprojekt Chip-Gruppe - Workshop, Ulm Juli 2000.
- [4] Christ, A.: Labwork Data Transmission: "Rigorous field numerical simulation of waveguide problems". Fachhochschule Offenburg, 2000.