



Medienkonzeption und Produktion  
Wintersemester 2007 / 2008

## Entwickler- Dokumentation

15. Januar 2008  
Version 1.0

Henrich, Lukas  
[lukashenrich@gmx.de](mailto:lukashenrich@gmx.de)

Würth, Alexander  
[alexanderwuerth@t-online.de](mailto:alexanderwuerth@t-online.de)

Prof. Hendrik Speck

Fachhochschule Kaiserslautern  
Amerikastr. 1  
66482 Zweibrücken  
Germany

# Inhalt

<b>1 Einleitung</b> .....	<b>3</b>
<b>2 Was ist XING?</b> .....	<b>3</b>
<b>3 Beschreibung des Produkts</b> .....	<b>3</b>
<b>4 Programmiersprachen und Systeme im Überblick</b> .....	<b>3</b>
<b>5 Architektur der Anwendung</b> .....	<b>4</b>
5.1 Software-Komponenten .....	4
5.2 Architektur .....	4
<b>6 Technischer Entwurf</b> .....	<b>5</b>
6.1 Überblick: Module .....	5
6.2 Crawler .....	5
6.3 Parser .....	5
6.4 Netzwerkgraph .....	6
6.5 Statistiken .....	6
<b>7 Datenbank-Entwurf</b> .....	<b>7</b>
7.1 Tabellen/Relationen .....	7
7.2 ER-Modell .....	8
<b>8 Software-Entwurf</b> .....	<b>8</b>
8.1 Methoden .....	9
8.2 Packet- und Verzeichnisstruktur .....	9
8.3 Verwendete JAVA-Bibliotheken .....	9
<b>9 Technische Daten</b> .....	<b>9</b>
9.1 Voraussetzungen .....	9
9.2 Datenbankverbindung einrichten .....	10
9.3 Datenbankstruktur erstellen .....	10
9.4 Datenbank füllen .....	10
9.5 Crawler einrichten .....	10
9.6 Verbindungsdaten CVS .....	10
<b>10 Kontakt</b> .....	<b>11</b>
10.1 Entwicklung .....	11
10.2 Betreuung .....	11

# 1 Einleitung

Durch die Explosion des Internets in den letzten Jahren und somit auch den Communities, welche sogenannte soziale Netzwerke bilden, existiert eine enorme Datenflut. XA soll ein OpenSource-Produkt darstellen, welches somit kostenlos nutzbar ist. Statistische Auswertungen und Analysen der Community XING, die mehrere Millionen Mitglieder hat, sollen mit der Anwendung möglich sein.

Das Projekt wurde im Rahmen der Veranstaltung „Medienkonzeption und Produktion“ des Studiengangs Digitale Medien an der Fachhochschule Kaiserslautern, Standort Zweibrücken erstellt.

## 2 Was ist XING?

[XING](#) ist eine webbasierte Plattform, welche ein sog. soziales Netzwerk in der Onlinewelt darstellt. XING hat derzeit ca. 5 Millionen Mitglieder und gehört somit zu den größeren Mitbewerbern in diesem Marktbereich. Die User können dort ihren eigenen digitalen Steckbrief mit einer Vielzahl an Eingabemöglichkeiten verwalten und Kontakte zu anderen Usern knüpfen. Eine Person kann also beispielsweise auch abfragen, „über wie viele Ecken“ sie eine andere Person kennt. Die Basic-Mitgliedschaft bei XING ist kostenlos. Gegen eine monatliche Gebühr kann auch wahlweise eine Premium-Mitgliedschaft erworben werden.



## 3 Beschreibung des Produkts

Die Software Xing Analyzer (XA) soll im Allgemeinen Daten des bekannten sozialen Netzwerks XING analysieren und visualisieren. Dabei können zum einen Kontaktbeziehungen der Benutzer ausgewertet und in einer ansprechenden Form visualisiert werden. Die Hauptfunktion stellt jedoch die statistische Auswertung personenbezogener Profildaten der einzelnen Benutzer dar. Auch die Darstellung dieser Daten wird durch graphische Elemente (wie z.B. Diagramme) unterstützt.

Die erfassten Daten von XING werden in einer MySQL-Datenbank gespeichert. Für alle Algorithmen zur statistischen Auswertung und die Datenbankzugriffe wird die Programmiersprache Java verwendet. Für das GUI der Anwendung kommen Techniken wie Java Swing oder Java AWT zum Einsatz.

## 4 Programmiersprachen und Systeme im Überblick

- Java

- HTML / CSS
- MySQL Datenbank

## 5 Architektur der Anwendung

### 5.1 Software-Komponenten

Bei der Anwendung XA sollte es sich ursprünglich um eine Webanwendung handeln. Da jedoch größere Probleme beim einbinden des Codes in ein Applet auftraten und diese in dem engen Zeitplan nicht mehr gelöst werden konnten, existiert die Anwendung jetzt zunächst als Desktop-Anwendung. Zunächst wird eine Datenbank (DB) benötigt, da die entsprechenden Daten der Seite dynamisch verwaltet werden sollen. Als Datenbank-Management-System kommt ein MySQL System zum Einsatz. Hauptgrund für diese Entscheidung war, dass dieses System kostenlos genutzt und relativ einfach installiert und gewartet werden kann. Die DB ist jedoch so entworfen, dass ohne Anpassungen bzw. Änderungen auf ein neues DBMS umgestellt werden kann.

Die Anwendung ist fast ausschließlich mit der Programmiersprache JAVA erstellt worden. Da es sich um eine Webanwendung handelt, wurde ein JAVA Applet eingesetzt. Zur Gestaltung der Website wurden darüber hinaus noch HTML, CSS und JavaScript benutzt.

Der Client-Rechner benötigt einen Webbrowser, um die Anwendung zu nutzen. Unterstützt bzw. getestet sind folgende Browser: Mozilla, Firefox, Netscape, Internet Explorer, Opera. Für den jeweiligen Browser muss zusätzlich noch eine Java Virtual Machine in Form eines Plugins installiert sein. Dies ist nötig, weil die Anwendung beim Client als Applet läuft.

### 5.2 Architektur

Die Anwendung wurde komplett nach dem Prinzip des Model-View-Control (MVC) Modells realisiert. Für die Umsetzung dieses Konzeptes bietet sich auf jeden Fall eine objektorientierte Programmiersprache an. JAVA wurde als Programmiersprache gewählt, weil durch die „starke“ Objektorientierung eine klare Trennung zwischen den einzelnen Schichten des Modells möglich ist. Des Weiteren bietet JAVA eine Vielzahl an Funktionen und Bibliotheken zum Umgang mit Datenbanken und Browsereinbettung. Weiter existieren einige gute Frameworks, die das MVC-Modell unterstützen.

Die Model-Schicht und Control-Schicht wurden mit Hilfe „reiner“ JAVA-Klassen umgesetzt. D.h. es wurden für diese Aufgaben auch keine bestimmten Frameworks benutzt. Für die View-Schicht wurde die API JAVA Swing verwendet.

## 6 Technischer Entwurf

### 6.1 Überblick: Module

- Daten sammeln
  - o Crawler
  - o Parser
- Netzwerkgraph
- Statistiken

### 6.2 Crawler

Der Sourcecode dieses Moduls befindet sich in dem Package `de.xa.crawler`. Der Crawler ist die Komponente, die benötigt wird, um die Daten von XING zu erfassen.

Vorgehensweise:

1. Bestimmte Anzahl von (zufälligen) Profilen über Google beziehen und in DB abspeichern (Java-Klassen: `de.xa.crawler.GoogleCrawler`, `Crawler`)
2. Weitere Profile aus den Kontaktbeziehungen der in der DB bereits gespeicherten Profilen (1.) erfassen (Java-Klassen: `de.xa.crawler.ContactCrawler`, `Crawler`)

Hinweis:

Um den Crawler nutzen zu können, muss zuvor ein Profil bei XING angelegt worden sein. Die Login-Daten müssen dem Crawler noch mitgeteilt werden (siehe „Technische Daten“). Wegen eines Sicherheitsmechanismus von XING, welcher das Anschauen von mehreren tausend Profilen innerhalb kurzer Zeit versucht zu verbieten, kann das crawlen von ca. 2500 Profilen zum Sperren des eigenen Profils führen. Eine Lösung, um diese Sperrung zu unterbinden, wurde leider noch nicht gefunden. Deshalb sind momentan nur ca. 7500 Datensätze in der DB.

### 6.3 Parser

Der Sourcecode dieses Moduls befindet sich in dem Package `de.xa.parser`. Der Parser filtert aus den "rohen" HTML-Seiten die Informationen heraus, die für die Anwendung benötigt werden. Auch werden die Daten so aufbereitet, dass sie in der entsprechenden DB abgelegt werden können.

Die Anwendung besitzt drei Parser:

1. Google-Parser (Java-Klasse: `de.xa.parser.google`)

- a. Parst eine Google Ergebnisseite nach Links auf XING Profildaten
- 2. Profile-Parser (Java-Klasse: `de.xa.parser`)
  - a. Parst eine XING Profildaten nach Userdetails wie (Name, Adresse, usw.)
- 3. Contact-Parser (Java-Klasse: `de.xa.parser.links`)
  - a. Parst eine XING Kontaktseite nach der Anzahl der Kontaktseiten

## 6.4 Netzwerkgraph

Der Sourcecode dieses Moduls befindet sich in dem Package `de.xa.io` und in der Klasse `de.xa.gui.ContactsView.java`. Im Netzwerkgraph werden die Kontaktbeziehungen eines Users (je nach dem, welcher Knoten angeklickt wurde) visualisiert.

## 6.5 Statistiken

Ein Teil des Sourcecodes dieses Moduls befindet sich in dem Package `de.xa.action`. Hinter diesem Modul verbirgt sich die Hauptfunktionalität der Anwendung. Es können eine Vielzahl von Statistiken angezeigt werden und diese dann mit verschiedenen Filtern eingegrenzt werden.

Die Klasse `de.xa.gui.StatisticsView.java` dient zur Anzeige der Ergebnisse in einem Diagramm (wahlweise Kuchendiagramm o. Balkendiagramm). Hier beschränkt sich die Ergebnismenge einer Statistik jedoch auf maximal 16 Treffer, damit eine vernünftige grafische Darstellung gewährleistet wird.

Eine ausführliche Präsentation aller Ergebnisse wird dem Benutzer dennoch geboten. Mit Hilfe der Klasse `de.xa.gui.StatisticsDetailView.java` werden diese in einer Liste, detailliert mit Angabe der prozentualen Verteilung, angezeigt.

Alle sinnvollen Variationen von Indikatoren sind in folgender Matrix beschrieben:

	Country	Province	City	Titel	Birthday	Branch	Size of Comp	Type of Comp	Status Time	Status Work
Country										
Province	X									
City	X	X								
Titel	O	O	O			O	O	O		O
Birthday										
Branch	O	O	O							
Size of Comp	O	O	O			O				
Type of Comp	O	O	O			O	O			
Status Time	O	O	O			O				
Status Work	O	O	O			O				

X = Angabe zwingend um Statistik anzuzeigen  
 O = Angabe optional um Statistik anzuzeigen

## 7 Datenbank-Entwurf

Als Datenbank-System kommt ein MySQL System zum Einsatz. Dieses läuft auf einem Server der Fachhochschule Kaiserslautern. Die DB ist so entworfen, dass ohne Anpassungen bzw. Änderungen, auf ein neues DBMS umgestellt werden kann. Dies wurde dadurch erreicht, dass nur Befehle aus dem SQL-Standard verwendet wurden.

### 7.1 Tabellen/Relationen

Die Tabellen der DB befinden sich in der 3. Normalform. Jede Tabelle besitzt ein eindeutiges Feld (ID), welches den Primärschlüssel der jeweiligen Tabelle darstellt.

Nachfolgend sind alle Tabellen des Systems mit Spaltennamen und Datentypen aufgelistet. Zusätzlich gibt es zu jeder Tabelle eine Beschreibung der einzelnen Felder. Eine genaue Beschreibung der Tabellenstrukturen ist im Anhang als DDL gegeben.

*Tabelle: main*

Diese Tabelle beinhaltet die Hauptdaten eines Users. Sie ist der Kern der DB-Struktur.

user_id	INTEGER	NOT NULL PRIMARY KEY
user_lastname	TEXT	NOT NULL
user_firstname	TEXT	NOT NULL
user_titel	TEXT	NULL
user_birthday	TEXT	NULL
user_adress_country	TEXT	NULL
user_adress_province	TEXT	NULL
user_email	TEXT	NULL
user_busi_adress_street	TEXT	NULL
user_busi_adress_zip	TEXT	NULL
user_busi_adress_city	TEXT	NULL
user_busi_adress_country	TEXT	NULL
user_busi_adress_province	TEXT	NULL
user_busi_email	TEXT	NULL
user_work_company_name	TEXT	NULL
user_work_branch	TEXT	NULL
user_work_position	TEXT	NULL
user_work_status_time	TEXT	NULL
user_work_size_of_comp	TEXT	NULL
user_kind_of_job	TEXT	NULL
user_image_url	TEXT	NOT NULL
user_work_status_work	TEXT	NULL

user_contacts_crawled	INTEGER	NOT NULL
user_profile_url	TEXT	NULL

*Tabelle: language*

In dieser Tabelle werden die Sprachen, die ein User angegeben hat, abgelegt.

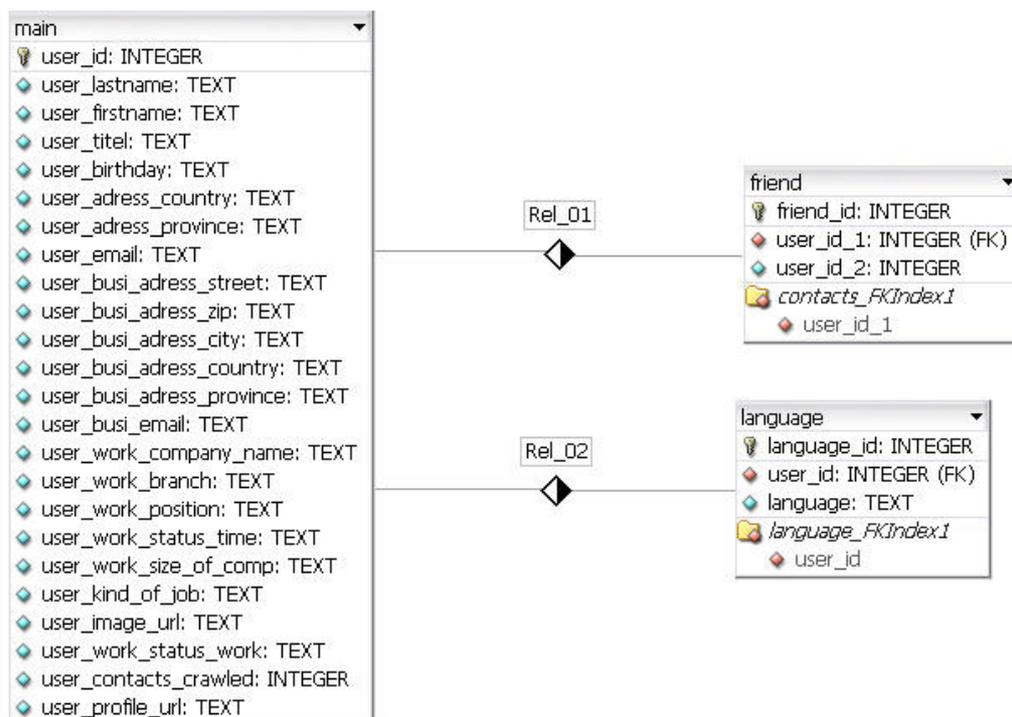
language_id	INTEGER	NOT NULL PRIMARY KEY
user_id	INTEGER	NOT NULL
language	TEXT	NULL

*Tabelle: friend*

Diese Tabelle dient dazu, die Freundschaftsbeziehungen zwischen den Usern abzuspeichern. Diese werden für die grafische Darstellung im Netzwerkgraph benötigt.

friend_id	INTEGER	NOT NULL PRIMARY KEY
user_id_1	INTEGER	NOT NULL
user_id_2	INTEGER	NOT NULL

## 7.2 ER-Modell



## 8 Software-Entwurf

Unter dem Punkt „Architektur“ wurde bereits die grobe Software-Architektur beschrieben. Die Anwendung besteht zum größten Teil aus JAVA-Klassen und läuft als Applet in einem Webbrowser.

## 8.1 Methoden

Die Methoden sind im Anhang als JAVA-Doc detailliert dokumentiert.

## 8.2 Packet- und Verzeichnisstruktur

Auf der JAVA-Ebene (Application-Source), auf der alle Klassen des MVC-Modells realisiert wurden, wurden die Klassen wie üblich in Paketen organisiert. Die erste Packet-Ebene beschreibt entweder eine Schicht des MVC-Modells, oder ein Modul der Webanwendung.

Die HTML Seiten der Website liegen in dem htdocs-Verzeichnis (Root-Verzeichnis für HTML, JSP, ... auf dem Server) des Tomcat-Webservers. Auch die Seiten, die für das Laden des Applets verantwortlich sind, sind dort abgelegt.

## 8.3 Verwendete JAVA-Bibliotheken

MySQL JDBC	Verbindung zur DB
Prefuse (beta)	Erstellung von Graphen
JFreeChart 1.0.6	Visualisierung von Statistiken
Commons HttpClient 3.1	Eigener Http-Client (benötigt für Crawler)
Commons Codec 1.3	<Commons HttpClient>
Commons Logging 1.1	<Commons HttpClient>

# 9 Technische Daten

## 9.1 Voraussetzungen

Hardware:

- Intel Pentium ab 1,5 GHz (oder vergleichbar)
- min. 512 MB Arbeitsspeicher
- 64 MB, 16 Bit Grafikkarte

- min. Auflösung 800 x 600 (besser mehr)
- 10 MB freie Festplattenkapazität
- Internetzugang (min. DSL 1000)

Software:

- Java Runtime Environment (JRE) 1.5, oder höher

## 9.2 Datenbankverbindung einrichten

Die Verbindungsdaten zur DB sind in einem Properties-File definiert. Im Projektverzeichnis unter `src/de/xa/properties/` existiert die Datei `access_mysql.properties` zur Einrichtung der Verbindungsdaten. Hier können Treiber, URL, User und Passwort angepasst werden.

## 9.3 Datenbankstruktur erstellen

Im Projektverzeichnis unter `src/de/xa/dbscript/` befindet sich ein SQL-Script `create_tables.sql` zum Aufbau aller Tabellen, Sequences, usw., die für die Anwendung benötigt werden. Mit `drop_tables.sql` können alle Tabellen gelöscht werden.

## 9.4 Datenbank füllen

Die Datenbank wird mit Hilfe des Crawlers gefüllt (s. nächsten Punkt).

## 9.5 Crawler einrichten

Um den Crawler nutzen zu können, muss zuvor ein Profil bei XING angelegt worden sein. Die Login-Daten werden dann in einem Properties-File definiert. Im Projektverzeichnis unter `src/de/xa/properties/` steht die Datei `general_settings.properties` dafür bereit.

## 9.6 Verbindungsdaten CVS

Für Entwickler:

Server: `xing-analyzer.cvs.sourceforge.net`  
Benutzer: <Login-Name bei sourceforge.net>  
Passwort: <Passwort bei sourceforge.net>  
Connection Type: `extssh`

Port: 22  
Path to Repository: /cvsroot/xing-analyzer

Anonym:  
Server: xing-analyzer.cvs.sourceforge.net  
Benutzer: anonymous  
Passwort: anonymous  
Connection Type: pserver  
Port: 2401  
Path to Repository: /cvsroot/xing-analyzer

## 10 Kontakt

### 10.1 Entwicklung

#### *Henrich, Lukas*

Anschrift: Friedhofstr. 23  
67715 Geiselberg  
Germany  
Telefon: +49 (0) 160 4143 237  
Email: lukashenrich@gmx.de

#### *Würth, Alexander*

Anschrift: Siemensstr. 2  
76751 Jockgrim  
Germany  
Telefon: +49 (0) 176 2472 7841  
Email: alexanderwuerth@t-online.de

#### *Homepage von XING Analyzer*

<http://www.xing-analyzer.de.vu/>

### 10.2 Betreuung

#### *Prof. Hendrik Speck*

Veranstaltung: Medienkonzeption und Produktion (WS 2007/2008)  
Studiengang: Digitale Medien

#### *Fachhochschule Kaiserslautern*

Amerikastr. 1  
66482 Zweibrücken  
Germany

<http://www.fh-kl.de/>