Data Mining Notre Dame

Principal Investigators: Vince Freeh, **Greg Madey**, Reene Tynan

Institut für Informatik

FU Berlin

13.03.09

Vorweg :: Greg Madey

Freie Universität Berlin

- Madey :: Woher ? **University of Notre Dame**

- Untersuchungen :: Welche ?
  - **Data Mining,** Forschung rund um **SourceForge.net**
    - **Network Analysis**
      - „small world phenomenon"
    - **Topological** Analysis of the OSS Developer Community
      - z.B. Verteilung von sozialen Positionen
    - **Temporal** Analysis
      - z.B. Änderung sozialer Positionen
  - **SRDA** (SourceForge Research Data Archive)

- Daten :: Woher ?
  - SourceForge.net HTML-Site **Crawling** 01.2001-05.2003
  - SourceForge.net Database **Snapshots** 11.2004-10.2008

Vorweg :: chronologische Reihenfolge

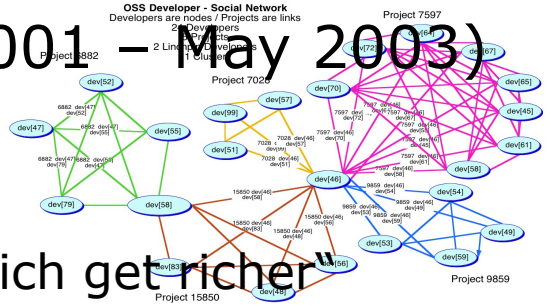
Freie Universität Berlin

- *Wie kam ich auf den Aufbau der gleich folgenden Folien*
  - Erster Versuch: ein gutes Paper finden
    - Fehlgeschlagen
  - Zweiter Versuch: ein Paper genannt bekommen
    - → Versucht zu verstehen
    - → Fehlgeschlagen
    - → Weitere Paper mit Details
      - → Versucht zu verstehen
      - → Fehlgeschlagen
      - → Weitere Paper mit Details
        - → Versucht zu verstehen
        - ...
    - → Fehlgeschlagen
    - Immer wieder Erklärungslücken in den Papern
  - → Dritter Versuch, diesmal **alle Paper** von Anfang an
    → und **in chronologischer Reihenfolge**

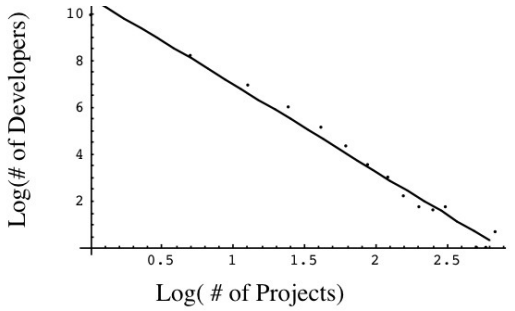
Vorweg :: 3 Phasen Übersicht

Freie Universität Berlin

- Phase 1 → Web-Site Crawling (ca. Jan 2001 – May 2003)
  - **Social Network Analysis**
    - Developer ↔ Project
    - preferential attachment of new nodes → „rich get richer"
    - diameter, clustering coefficient, degree distribution
    - Agent-Based Modeling and Simulation → Java + Swarm



- Phase 2 → SourceForge 2003 data dump
  - Onion Model
  - **Analysis** of **activities**
  - Developer roles
- Phase 3 → Data dumps 2003 – 2008
  - User/project pair comparison
    - 29 activities (21 + 8)
    - **Global** Social Positions **vs.**
    - **Temporal** Social Positions



Phase 1/3 :: Von den Daten zum Netz SourceForge.net durchlaufen

Freie Universität Berlin

- 1. Webcrawling
  - SourceForge durchgehen
    - Web Crawler
  - Projekte + Developer Daten sammeln
    - Perl Skript
  - in Textdatei speichern

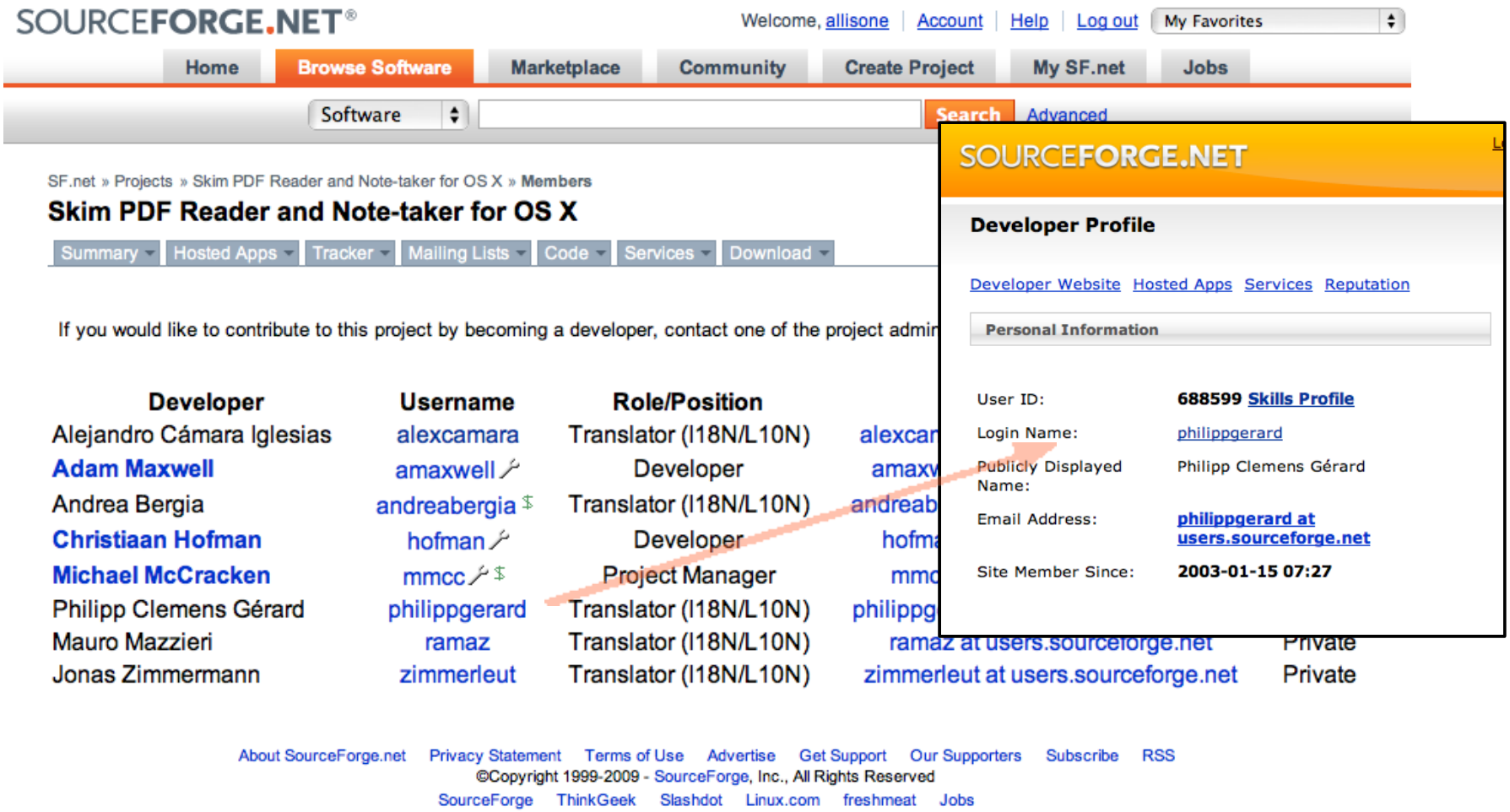
8001|dev378
8001|dev8975
8001|dev9972
8002|dev27650
8005|dev31351
8006|dev12509
8007|dev19395
8007|dev4622
8007|dev35611
8008|dev7698

→ **Bild** nächste Folie

Phase 1/3 :: Von den Daten zum Netz SourceForge.net durchlaufen

Freie Universität Berlin

Bsp: http://sourceforge.net/project/memberlist.php?group_id=192583



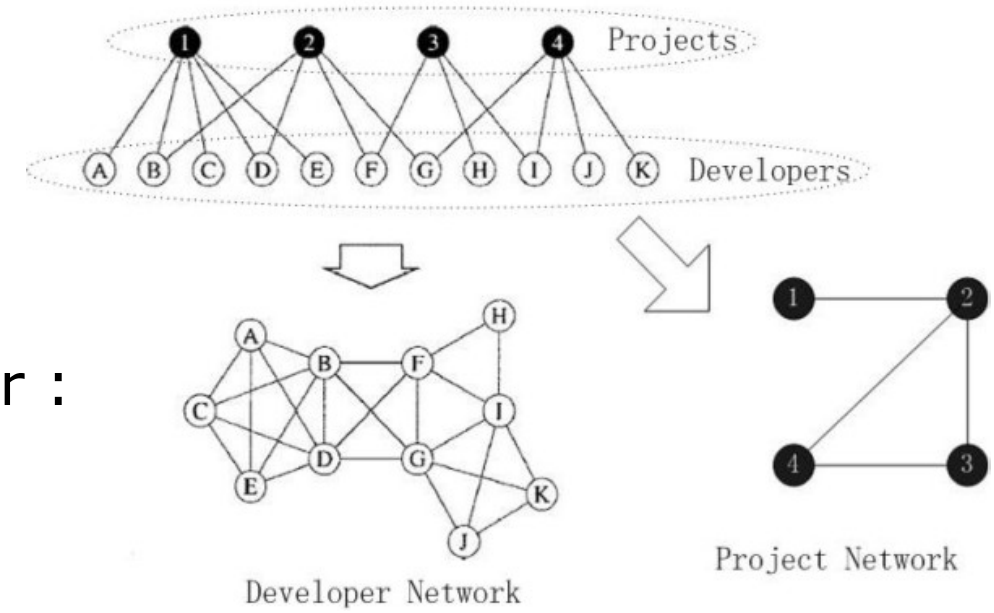
Phase 1/3 :: Social Network Analysis

Freie Universität Berlin

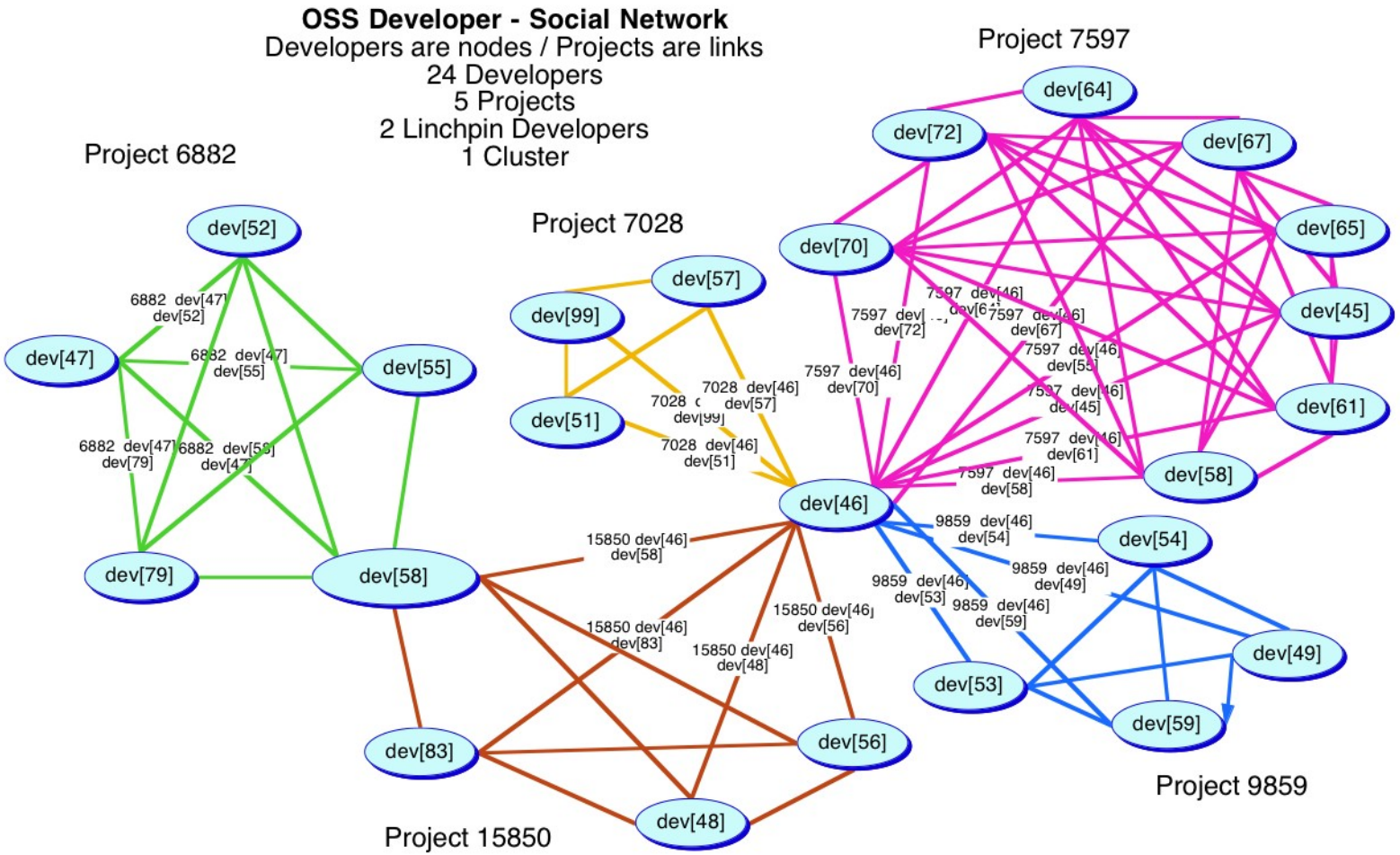
- 2. Text → Oracle Datenbank

```
8001|dev378
8001|dev8975
8001|dev9972
8002|dev27650
8005|dev31351
8006|dev12509
8007|dev19395
8007|dev4622
8007|dev35611
8008|dev7698
```

- 3. Gegeben:
  - Entwickler in Relation mit Projekten
- Daraus herstellbar:
  - Bipartiter Graph
- Bipartiter Graph wandelbar :
  - Entwickler Netzwerk
  - Projekt Netzwerk



Phase 1/3 :: Ein Cluster



Phase 1/3 :: Social Network Analysis Vermutungen

Freie Universität Berlin

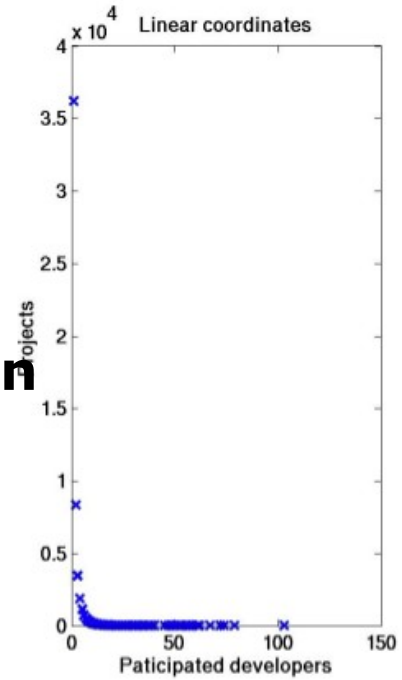
- Erste Erkenntnisse
  - Die **10 höchstgerankten Projekte** von SourceForge.net
    - Haben **im Schnitt 20x mehr Entwickler** / Projekt als der Rest
  - Graph ist also kein *random graph*
    - dachte man sich auch schon vorher so
  - Wahrscheinlich Graph mit ***preferential attachment***
    - → weitere Datenanalyse

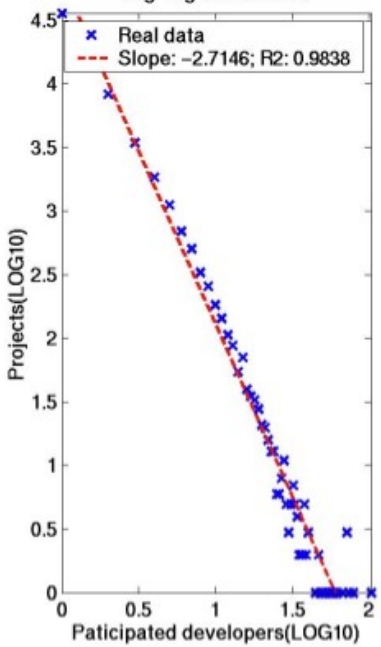
SourceForge rates projects by activity, such as downloads, updates, and page hits. This allows one to find distinguishing characteristic of top projects. Looking at the top 10 projects for the month of February 2002, we find that there is an average of 29 developers per project, 20 times the SourceForge average. Also these developers belong to 25% more projects than the overall population. Finally, 70% of the developers on top projects belong to exactly one project (versus 80% overall), but the maximum is only 12.

Phase 1/3 :: Social Network Analysis Ergebnisse

Freie Universität Berlin

- Power-Law Distributions
  - Es gibt seeeehr **viele Projekte mit nur einem Entwickler**
  - Und sehr **wenig Projekte**, **mit** seeeehr **vielen Entwicklern**
  - auch andersrum



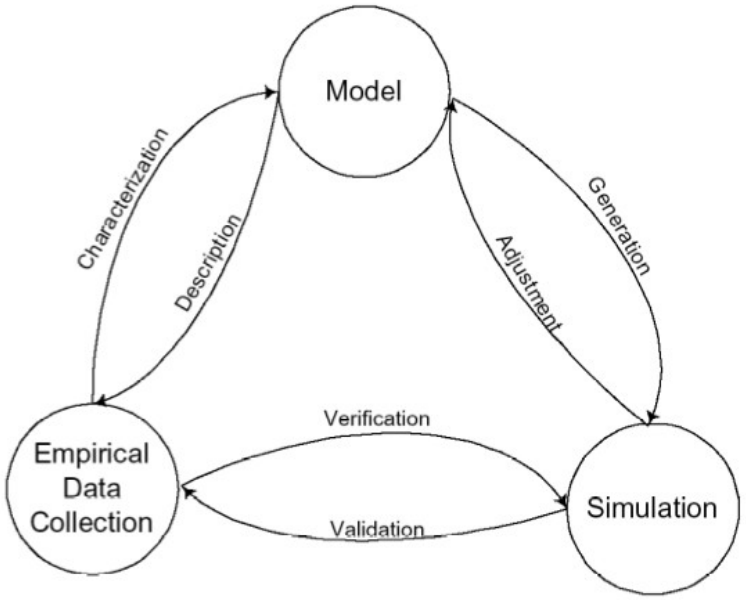


- Ausserdem
  - Diameter = 6-8 [2003 Gao, Analysis and modeling of the open source software]
    - → Informationen verbreiten sich schnell im Netz
  - Clustering Coefficient = 0.7
    - Das heißt Knoten haben in ihren Cliquen Kanten zu 70 % der Cliquenmitglieder → **Tafelbild**
    - Random Network wären so circa 20 %

Phase 1/3 :: Simulationen

Freie Universität Berlin

- Vermutung
  - Soziales Netzwerk = Graph mit preferential attachment √
  - User untersch. Fitness ← („young upstart" phenomem) ?
- Evolutions-Simulationen
  - 4 Modelle
    - ER
    - BA
    - BA+Fitness
    - BA+dyn. Fitness



- Ziel:
  - Modell finden, möglichst identisches Abbild von SF.net
  - genaues Verständnis zu erlangen
  - Theorien bestätigen

Phase 1/3 :: Simulationen

Freie Universität Berlin

- Simulationen vs. empirische Daten
  - Verhältnis Entwickler zu Projekten
    - (obiger log log Graph)
  - Network Diameter
    - (Small-World-Phenomenon)
  - Clustering Coefficient
    - (wie gut verbunden)
  - Degree Distribution
    - (viele Projekte haben wenig Developer)
  - Cluster Distribution
    - (viele Cluster klein, wenig Cluster groß)
  - Average Degree
    - (Kanten pro Knoten im Schnitt)

Phase 1/3 :: Simulationen :: Ergebnis

Freie Universität Berlin

- ER Model
  - voll daneben
    - wie vermutet alle Werte „falsch“, außer Clustering Distribution
- BA Model
  - Treffer bei alle topologischen Eigenschaften des Netzwerks
    - diameter bei circa 7
    - clustering coefficient circa 0.7
  - Aber
    - „young upstart“ nicht beobachtbar
- BA Model + (constant) Fitness
  - Wieder Treffer
  - Aber
    - Nicht in der Lage, Entwicklung von Individuen zu reproduzieren

Phase 1/3 :: Simulationen :: Ergebnis

Freie Universität Berlin

- BA Model + dynamische Fitness
  (Am Anfang fit, langsam träger)

  - Ergebnis: BINGO !!!
    - gleiche topologische Ergebnisse, wie in
      - BA
      - BA + (constant) Fitness

      → ähnlich den empirischen Daten
    - und
      - Beobachtung variierender anfänglicher Power der Projekte und User
      - Mit zunehmendem Alter → Beruhigung und Stabilisierung

      → ähnlich den emprischen Daten

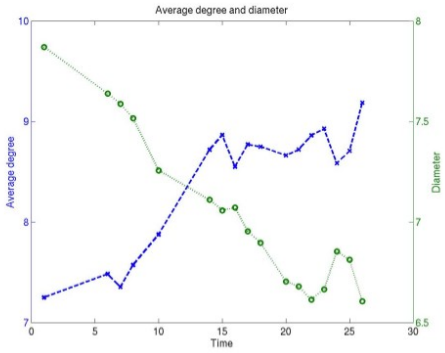
Phase 1/3 :: Zusammenfassung

Freie Universität Berlin

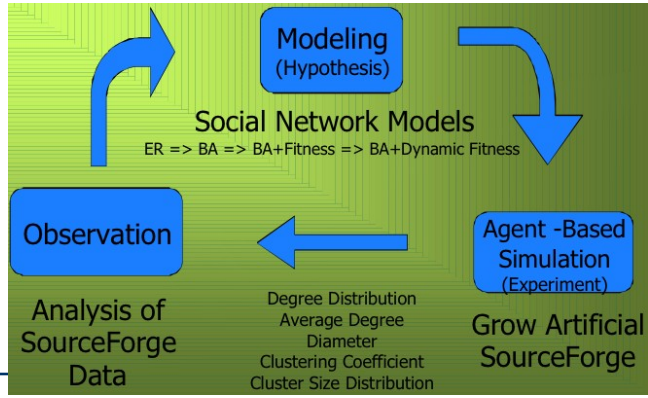
- Zusammenfassung dieser verwirrenden Informationen
  - Daten Sammlung
    - welche Entwickler in welchen Projekten

```
8001|dev9972
8002|dev27650
8005|dev31351
8006|dev12509
```

  - Netzwerk Analyse → Graphen-Theorie-Zeug



  - Simulationsmodelle Entwickeln
    - ER
    - BA, BA+fitness, BA+dynamic fitness
  - Vergleich mit emprischen Daten
    - ausschlaggebenden Faktoren verstehen, warum OS erfolgreich

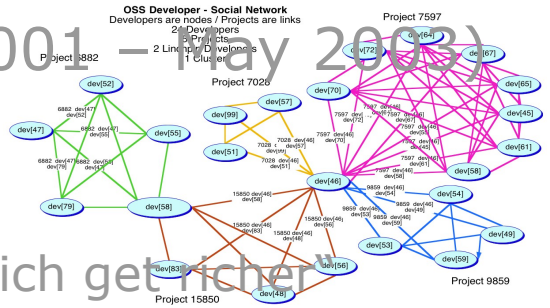


Luft holen !!!


COLD CASE SUSPECT?
RODNEY
STANGER
101
146
EARLY WARNING ALERT
GOODWIN COLLEGE
STANLEY-WHITMAN HSE.-FARMINGTON CLOSED
STONE ACAD. - HAMDEN NO EVE. CLASSES
STONE ACAD. - WTBY. NO EVE. CLASSES
YouTube

Gliederung

- Phase 1 → Web-Site Crawling (ca. Jan 2001 – May 2003)
  - **Social Network Analysis**
    - Developer ↔ Project
    - preferential attachment of new nodes → „rich get richer“
    - diameter, clustering coefficient, degree distribution
    - Agent-Based Modeling and Simulation → Java + Swarm



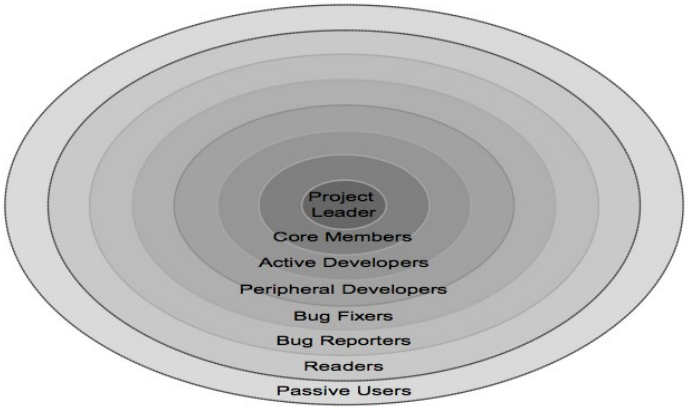
- Phase 2 → SourceForge 2003 data dump
  - Onion Model
  - **Analysis** of **activities**
  - Developer roles

Phase 2/3 :: Vom OnionModel zum...? Rollen und deren Verteilung

Freie Universität Berlin

- 2002 Rollenverteilung nach Onion Model → Rollenzahl = 8

  [Nakakoji et al. (auch Kishida), „Evolution Patterns of Open-Source Software Systems and Communities“]



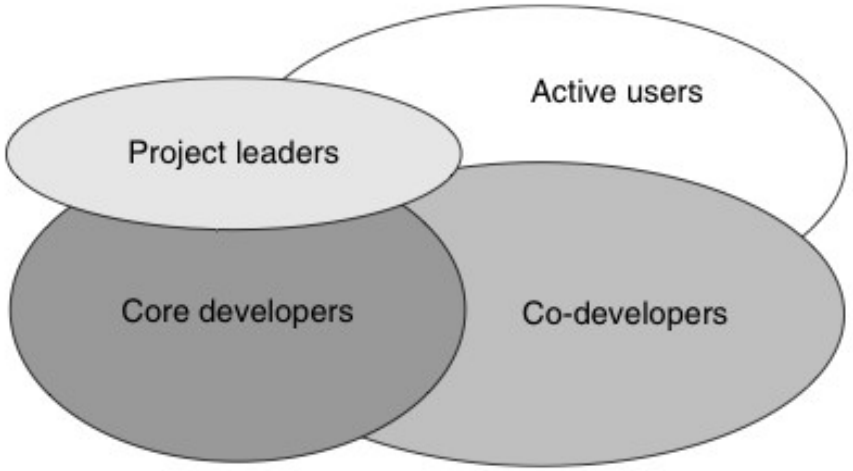
- 2003 Rollen modiziert / reduziert auf 6

  [Xu N., „An Exploratory Study of Open Source Software Based on Public Project Archives”]

  - Passive User, Active User,
  - Peripheral Deveoloper, Central Developer,
  - Core Developer, Project Leader
- 2004 Jin Xu und Greg Madey reduzieren auf 5
  - Passive User, Active User,
  - Co-Developer,
  - Core Developer, Project Leader

Phase 2/3 :: … Bubble Model ?
Rollen und deren Verteilung

- Project Leader
  - Project Admin
  - Vision, Direction of Project
- Core Developer
  - Extensivly contribute
  - Manage CVS releases
  - Coordinate Co-Dev
- Co-Developer
  - fix bugs
  - add features
  - provide support
  - write documents
- Active Users
  - discover bugs
  - suggest features
  - exchange information
  - no code modifying



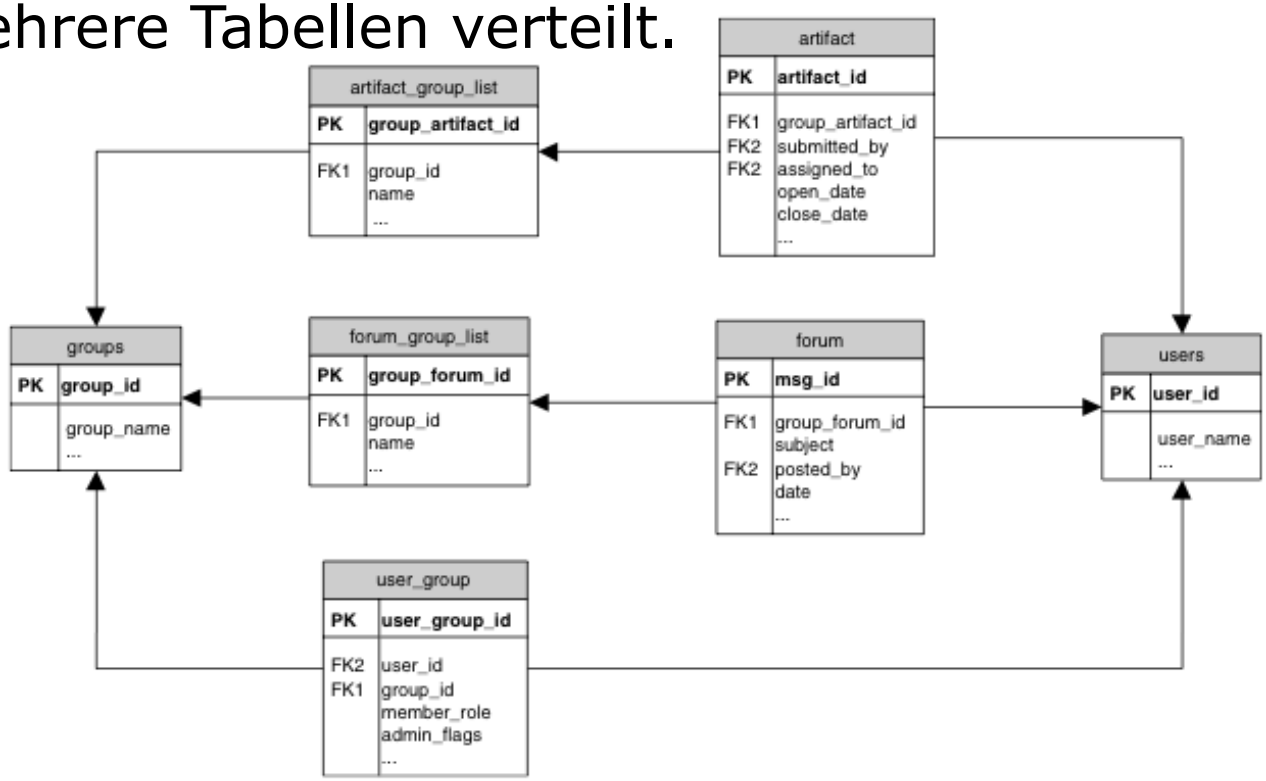
Man muss nicht erst Activer User sein, um Co-Developer zu werden. Und man muss als Co-Developer nicht unbedingt auch Bugs entdecken.

↔ Onion Model suggeriert anderes

Phase 2/3 :: Ziel: Rollen Verteilung
Problem: Daten zerstreut

Freie Universität Berlin

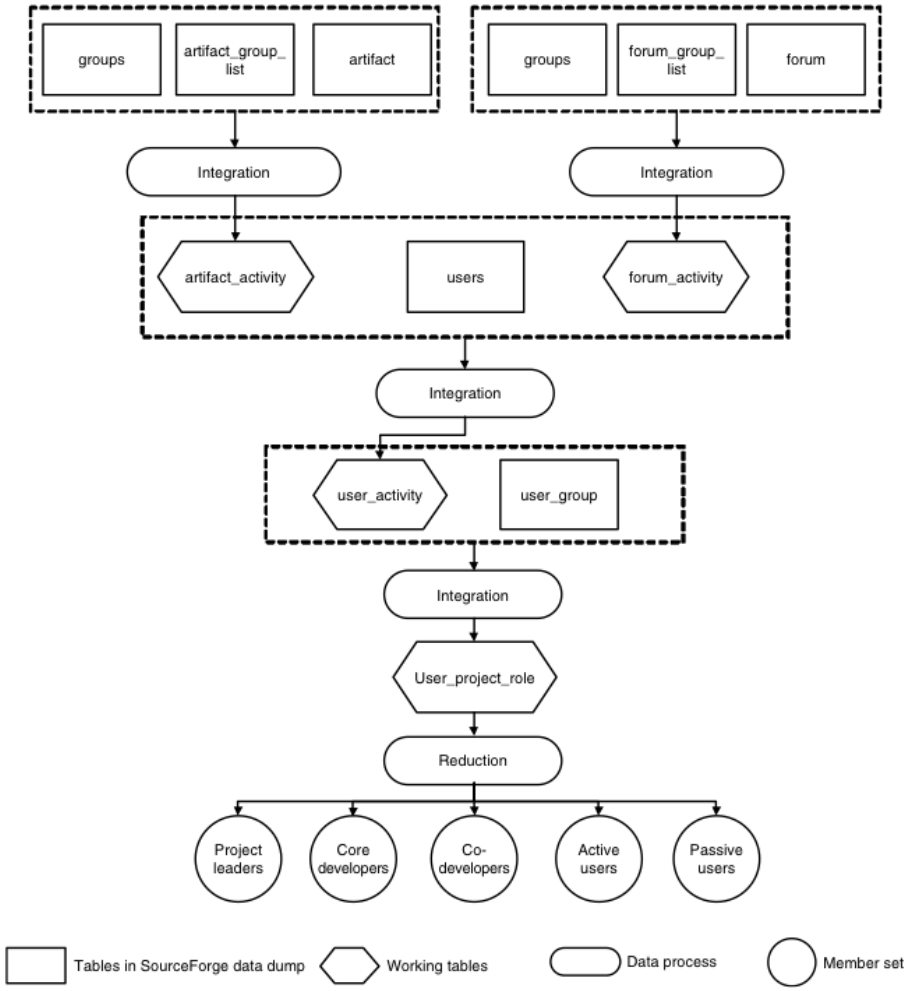
- Ziel nun: Prozentuale Verteilung der einzelnen Rollen untersuchen
- Problem: Project Leader und Core Developer bereits bei SF gelistet, aber Co-Developer, Aktive und Passive User über mehrere Tabellen verteilt.



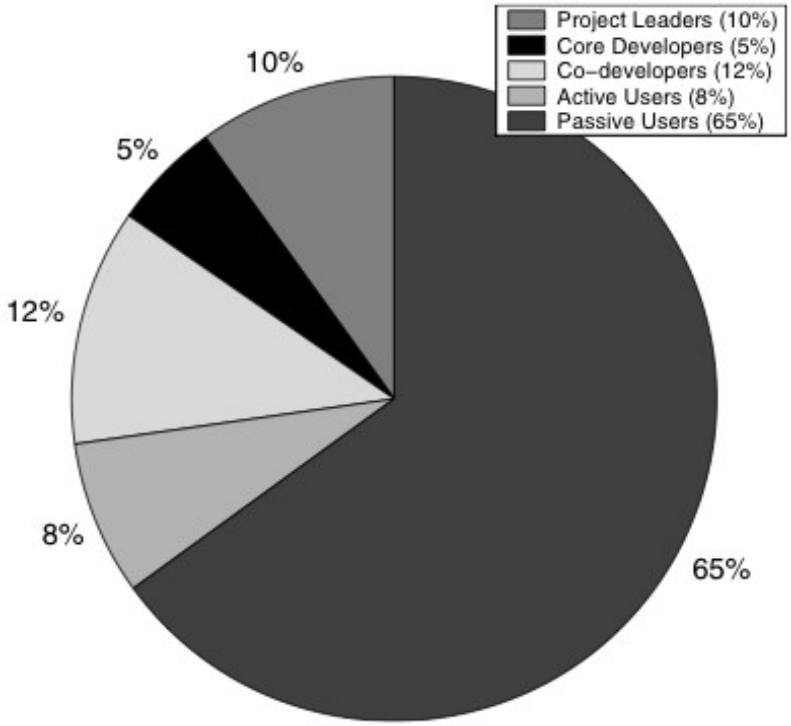
Phase 2/3 :: Ziel: Rollen Verteilung
Lösung: Integration und Reduktion

- Lösung: 3 Schritte
- Kombinieren der Artifakt- und Forum Tabellen
  - = artifact_activity
  - = forum_activity
- kombiniere dann mit *users*
  - = user_activity
- diese dann mit *user_group*
  - = user_project_role

- Mit einer Datenreduktion wurden dann diese Daten auf 5 Sets entsprechend der Rollen verteilt



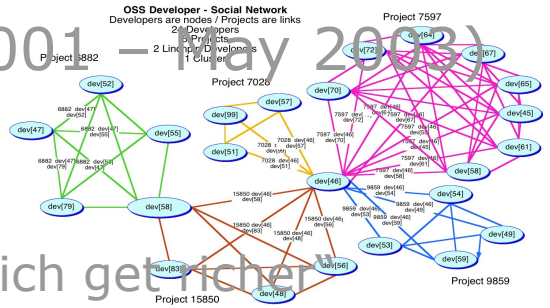
Phase 2/3 :: Ergebnis: User Rollen Verteilung



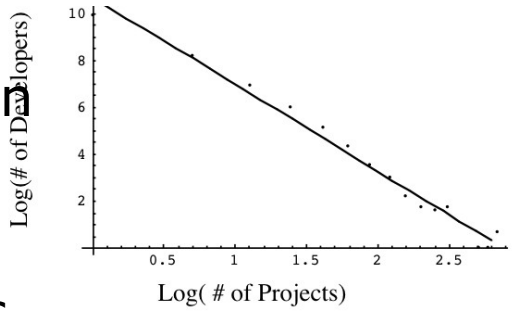
- siehe erstmal Bild
- interessant auch
  - kleine Projekte
    - 47,8 % Project Leader
    - 20,6 % Core Developer
  - große Projekte
    - 3,6 % Project Leader + Core Dev.
    - 55,8 % Co-Developer
    - 40,6 % Active User
  - Diameter
    - mit Co-Dev und Act. User = 3
    - vs. nur Lead+Core = 6-8

Gliederung

- Phase 1 → Web-Site Crawling (ca. Jan 2001 – May 2003)
  - **Social Network Analysis**
    - Developer ↔ Project
    - preferential attachment of new nodes → „rich get richer"
    - diameter, clustering coefficient, degree distribution
    - Agent-Based Modeling and Simulation → Java + Swarm



- Phase 2 → SourceForge 2003 data dump
  - Onion Model
  - **Analysis of activities**
  - Classify developer
- Phase 3 → Data dumps 2003 – 2008
  - **Clustering** by user/project pair comparison
    - by activities (21 + 8)
    - Global(all time) **Social Positions** vs.
    - Temporal(1st,2nd,... month) Social Positions



Phase 3/3 :: Übersicht

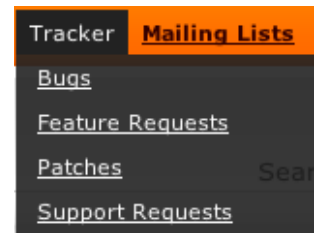
Freie Universität Berlin

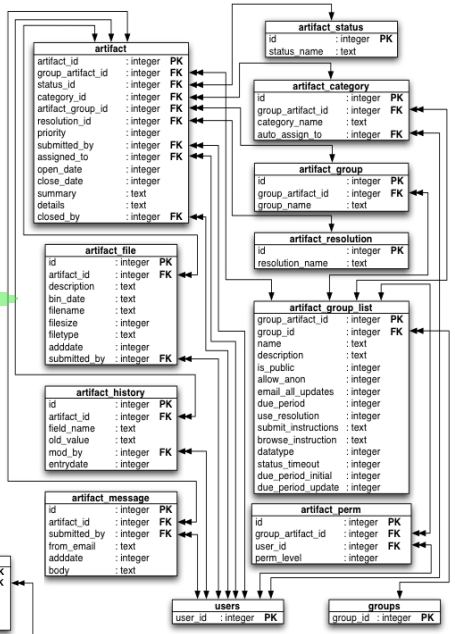
- Datenbestand auf dem Gearbeitet wird:
  - DB Dumps von Existenzbeginn von SF.net bis Juni 2005 ??
- Daten auf denen gearbeitet wird
  - Activity Data
    - 21 Typen aus dem SF.net back-end
    - +8 Typen aus CVS Repositories der Projekte
- Untersuchungen:
  - Temporale Analyse Sozialer Positionen
  - Globale Analyse Sozialer Positionen

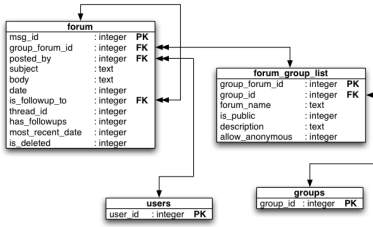
Phase 3/3 :: Activity Typen

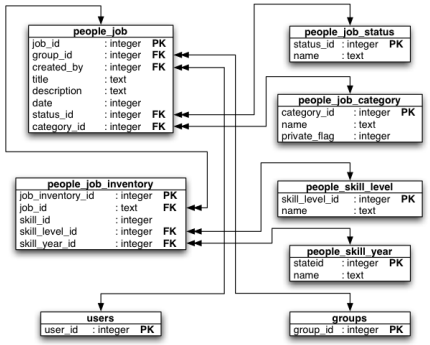
Freie Universität Berlin

	Activity Type	Activity Description
Artifact Activity	submit bug(1)	Person submits a new bug report.
	assign bug(2)	Bug report is assigned to person.
	submit support request(3)	Person submits a new support request.
	assign support request(4)	Support request is assigned to person.
	submit patch(5)	Person submits a new patch.
	assign patch(6)	Patch is assigned to person.
	submit feature request(7)	Person submits a new feature request.
	assign feature request(8)	Feature request is assigned to person.
	submit todo(9)	Person submits a new to-do item.
	assign todo(10)	To-do item is assigned to person.
	submit other artifact(11)	Person submits an artifact that is not one of the predefined categories of bug report, support request, patch, feature request, or to-do item.
	assign other artifact(12)	Uncategorized artifact is assigned to person.
Forum Activity	new forum message(13)	Person posts a new forum message.
	followup forum message(14)	Person posts a forum message that is a followup to an existing forum message.
Project History Activity	modify project(15)	Person makes an administrative modification to the project; the modification is uncategorized, but they are typically tasks like adding/removing members, changing permissions, updating project settings, etc.
File Release Activity	file release(16)	Person posts a new file release; this is typically associated with releasing a new version of the software to the public.
Project Task Activity	new project task(17)	Person creates a new project task.
	assigned project task(18)	A project task is assigned to person.
	modify project task(19)	Person modifies an existing project task.
Document Actity	create document(20)	Person creates a new document.
Job Activity	create people job(21)	Person posts a new job; these are similar to help-wanted ads where a project is looking for somebody with particular skills.
CVS Activity	checkout source code(22)	Person checks out source code from CVS repository.
	export source code(23)	Person exports source code from CVS repository.
	release source code(24)	Person releases check out of source code from CVS repository.
	tag source code(25)	Person tags source code in the CVS repository with a label.
	add source code file(26)	Person adds a new source code file to the CVS repository.
	remove source code file(27)	Person removes a source code file from the CVS repository.
	modify source code file(28)	Person commits a source code modification to the CVS repository.
	update source code(29)	Person updates local checked out source code with any changes in CVS repository.









Phase 3/3 :: Untersuchung der Sozialen Positionen

- Clustering, um soziale Positionen zu ermitteln

- Wie ? → Novel Algorithmus auf User/Project Pairs
  - User/Project Pair vergleichen
  - Statistical Test auf Verteilung der Variablen
    - Sind zwei Paare verschieden werden sie ausgesondert
    - sonst läuft der Algorithmus weiter
  - Dadurch werden alle Paare rausgeschmissen die unterschiedlich sind
  - Die restlichen müssen mit bestimmtem Confidence Faktor gleich sein und bilden also ein Cluster
  - Das wird solange wiederholt bis wir alle Cluster zusammen haben

Phase 3/3 :: Globale Untersuchung der Sozialen Positionen

Freie Universität Berlin

- Mit obigen Algorithmus und **Vergleich aller User/Projekt** Paare vom Beginn von SourceForge.net bis 2005 entsteht
- Der Novel Algo. $\longrightarrow$ 45 Cluster
- Nach Manueller Betrachtung $\longrightarrow$ 7
  - 6 Cluster = je eine soz. Pos.
  - 39 zu Sw Dev.

Social Position	Description	Size
Software User	The largest cluster with the primary activities of posting new forum messages, followup forum messages, and checkout out source code.	111889
Project Administrator	The second largest cluster with the primary activity of making project modifications; the project administrator also performs file releases, but most other activities are relatively minor or non-existent.	93199
Software Developer	Primary activities are source code operations like checking out source code, add/remove source code files, modify source code, and update source code. The social position contains 39 clusters all with different relative proportions of the source code operations, and some software developers have significant levels of project modification and file release activities.	47495
Task Management	Significant usage of the project task management provided by SourceForge.net.	2181
Bug Reporter	Significant bug reporting activity with a slight amount of features requests, support requests, and patches.	1138
Feature Requester	Primary activity was submission of feature requests but also has a significant amount of bug reporting.	370
Handyperson	The handyperson has significant activity for many different activity types including source code modifications, bug reporting, project modifications, file releases, and project tasks.	217
Not Categorized	The remaining very small clusters that were not analyzed.	14818
	Total user/project pairs	271307

Phase 3/3 :: Temporale Untersuchung der Sozialen Positionen

Freie Universität Berlin

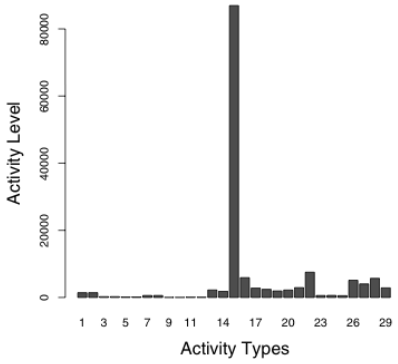
- Selbiges wie eben, aber nun Vergleich von **User/Projekt Paaren, zu Beginn jeweiliger User Existenz** (also dem Monat an dem Sie sich registrieren) bis zum 4ten Monat
- Erster Monat normal
- Zweiter Monat
  - Proj. Admin gone
  - Drop User/Proj Pairs.
  - Drop Message Poster
- Dritter Monat
  - Handyperson dominant

Social Position	Description	Month 0	Month 1	Month 2	Month 3
Project Administrator	Primary activity of making project modifications; other activities are relatively minor or non-existent.	86951	0	0	0
Message Poster	Primary activity of posting new or followup forum messages; other activities are relatively minor or non-existent.	96052	7315	0	0
Software Developer	Primary activities are source code operations like checking out source code, add/remove source code files, modify source code, and update source code. Multiple clusters with different relative proportions for the source code operations; some software developers have significant levels of project modification and file release activities.	67488	32126	21054	18239
Release Management	Primary activity of file release but also significant project modification activity.	11700	7227	0	0
Task Management	Significant usage of the project task management provided by SourceForge.net.	1775	0	0	0
Handyperson	The handyperson has significant activity for many different activity types including source code modifications, bug reporting, project modifications, file releases, and project tasks.	1768	120	14050	10709
Not Categorized	The remaining very small clusters that were not analyzed.	3066	1638	1712	1611
	Total user/project pairs	268800	48426	36816	30559

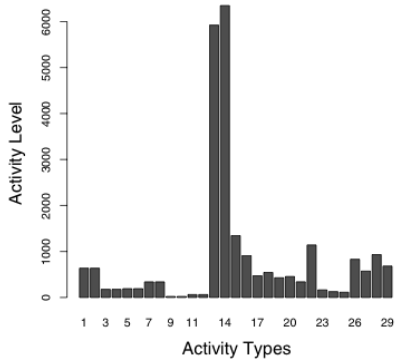
Phase 3/3 :: Einschub Beispiel
Soziale Pos.en mit Activity Verteilung

Freie Universität Berlin

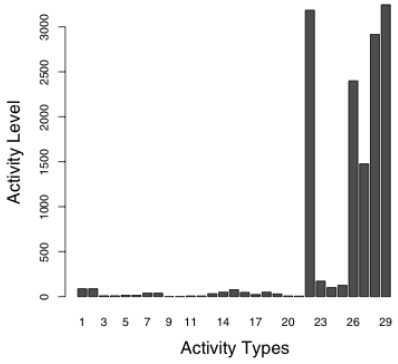
	Activity Type	Activity Description
Artifact Activity	submit bug(1)	Person submits a new bug report.
	assign bug(2)	Bug report is assigned to person.
	submit support request(3)	Person submits a new support request.
	assign support request(4)	Support request is assigned to person.
	submit patch(5)	Person submits a new patch.
	assign patch(6)	Patch is assigned to person.
	submit feature request(7)	Person submits a new feature request.
	assign feature request(8)	Feature request is assigned to person.
	submit todo(9)	Person submits a new to-do item.
	assign todo(10)	To-do item is assigned to person.
	submit other artifact(11)	Person submits an artifact that is not one of the predefined categories of bug report, support request, patch, feature request, or to-do item.
	assign other artifact(12)	Uncategorized artifact is assigned to person.
Forum Activity	new forum message(13)	Person posts a new forum message.
	followup forum message(14)	Person posts a forum message that is a followup to an existing forum message.
Project History Activity	modify project(15)	Person makes an administrative modification to the project; the modification is uncategorized, but they are typically tasks like adding/removing members, changing permissions, updating project settings, etc.
File Release Activity	file release(16)	Person posts a new file release; this is typically associated with releasing a new version of the software to the public.
Project Task Activity	new project task(17)	Person creates a new project task.
	assigned project task(18)	A project task is assigned to person.
	modify project task(19)	Person modifies an existing project task.
Document Actity	create document(20)	Person creates a new document.
Job Activity	create people job(21)	Person posts a new job; these are similar to help-wanted ads where a project is looking for somebody with particular skills.
CVS Activity	checkout source code(22)	Person checks out source code from CVS repository.
	export source code(23)	Person exports source code from CVS repository.
	release source code(24)	Person releases check out of source code from CVS repository.
	tag source code(25)	Person tags source code in the CVS repository with a label.
	add source code file(26)	Person adds a new source code file to the CVS repository.
	remove source code file(27)	Person removes a source code file from the CVS repository.
	modify source code file(28)	Person commits a source code modification to the CVS repository.
	update source code(29)	Person updates local checked out source code with any changes in CVS repository.



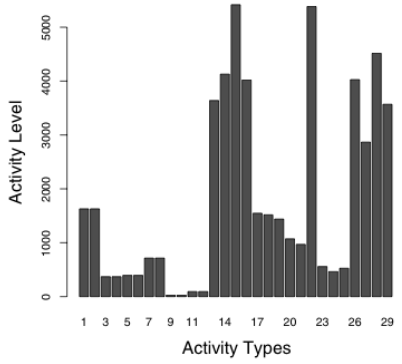
(a) Project Administrator



(b) Message Poster



(c) Software Developer



(d) Handyperson

Phase 3/3 :: Diskussion

Freie Universität Berlin

- Im gesamten Sozialen Netzwerk sind 40 % der Individuen Software User
  - führen Aktivitäten in scheinbar gröberen Abständen aus
    - Checken Code aus, lange Zeit später Nachrichten im Forum z.B.
    - Oder Vice Versa
- Die Handyperson (Der Alleskönner/-macher)
  - im globalen sozialen Netzwerk unbedeutend zu sein
  - aber Hauptrolle, wenn über die Zeit betrachtet.
- soziale Rollen bleiben nicht gleich
  - Aktivitäten und Verantwortungen ändern sich mit der Zeit

Phase 3/3 :: Untersuchung der Sozialen Positionen Diskussion

Freie Universität Berlin

- Der Wechsel
  - von z.B. Message Poster
  - zu Handyperson oder Softwaredeveloper
- Scheint Schlüsselereignis
  - um Aktivitätsniveau überersten Monate hinaus zu halten
- Einige Soziale Positionen
  - sind gleich im temporalen und dem globalen Netzwerk
  - einige existieren in dem einen, aber nicht im anderen
- Die Rollen Projekt Administrator und Software Developer
  - Erinnern zwar an „closed source" Entwicklung
  - jedoch scheint:
    - Möglichkeit Aufgaben aussuchen/schnell wechseln nur in OSS
    - → Open Source effizienter und schnell anpassbar

Vielen Dank!

Weiteres SF.net Crawl-Mining

Freie Universität Berlin

- 2003
  - Erstes mal erwähnt Project Statistics
    - Downloads, CVS Activities, Page Views, Bugs
    - Versuch z.B.: Downloads an Hand anderer Features vorauszusagen

Table 1. Project statistics

project ID	lifespan	rank	page views	downloads	bugs	support	patches	all trackers	tasks	cvs
1	1355 days	31	12,163,712	71,478	4,160	46,811	277	52,732	44	0
2	1355 days	226	4,399,238	662,961	0	53	0	62	0	14,979
3	1355 days	301	1,656,020	1,064,236	364	35	15	421	0	12,263
7	1355 days	3322	50,257	20,091	0	0	0	12	0	0
8	1355 days	2849	6,541,480	0	17	1	1	26	0	13,896