

ソフトウェア開発を支援するためのSocio- Technical Information Space(STIS)

葉 雲文@SRA

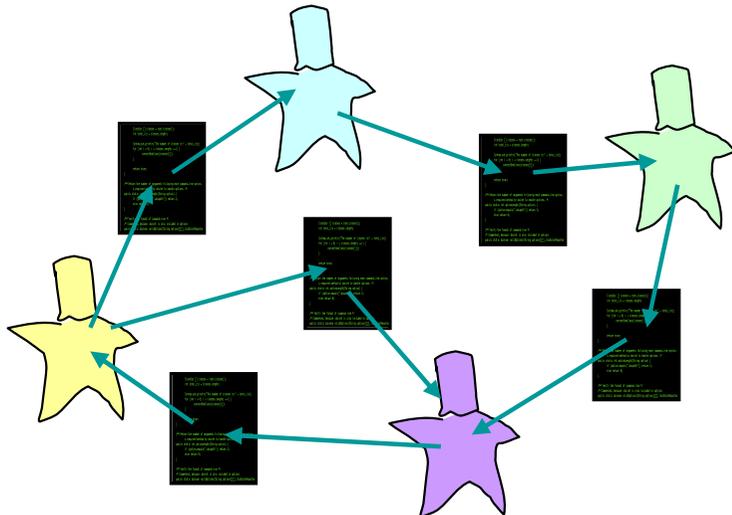
SS2009 WG1

知識労働としてのソフトウェア開発

- 農業労働
 - なんにもないところから物質を生産する
- 工業労働
 - 複数の物質を別の物質に変換させる
- 知識労働
 - なんにもないところから物質的な存在ではない新しい体験を作る
 - 知識 → 知識アーティファクト
 - 知識収集と知識創出の繰り返し

知的協調作業としてのソフトウェア開発

- 複数の人手の総和から複数の頭脳の相乗へ
 - Division of Labor vs Distributed Cognition(分散認知)
- 開発に必要な知識が分散している
 - ドメイン知識
 - プログラミング知識
 - 道具知識(開発環境、OS、API)
 - 開発中のシステムに関する知識

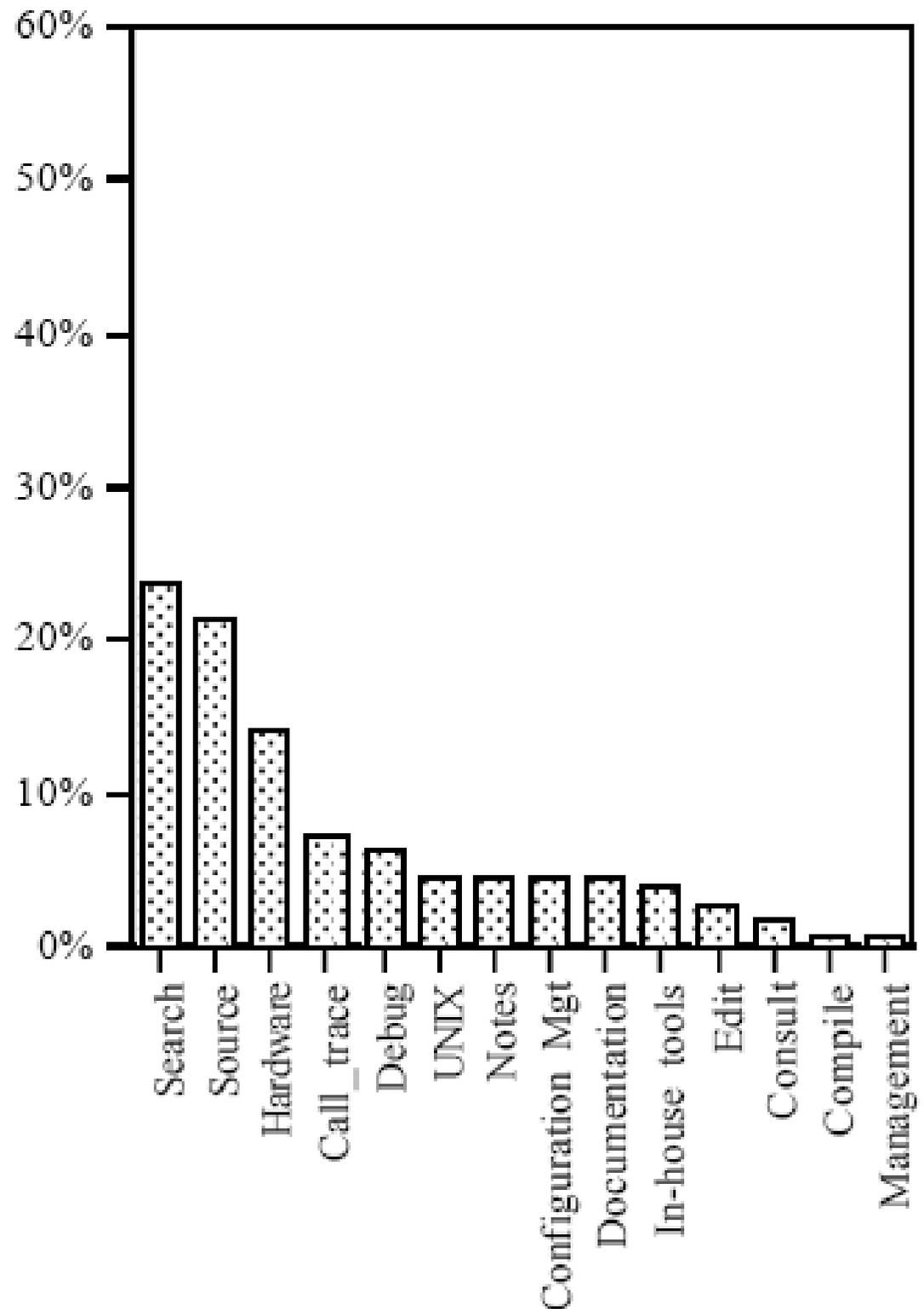


開発者観察

あるベテラン開発者の14日間の開発活動を観察した結果（合計 157 イベント）

サーチ 37回
ソースコード閲覧 33回

•Singer, J., Lethbridge, T., Vinson, N., and Anquetil, N., "An examination of software engineering work practices," in *Proceedings of the 1997 conference of the centre for advanced studies on collaborative research (cascon'97)*, 1997, pp. 21 pages.

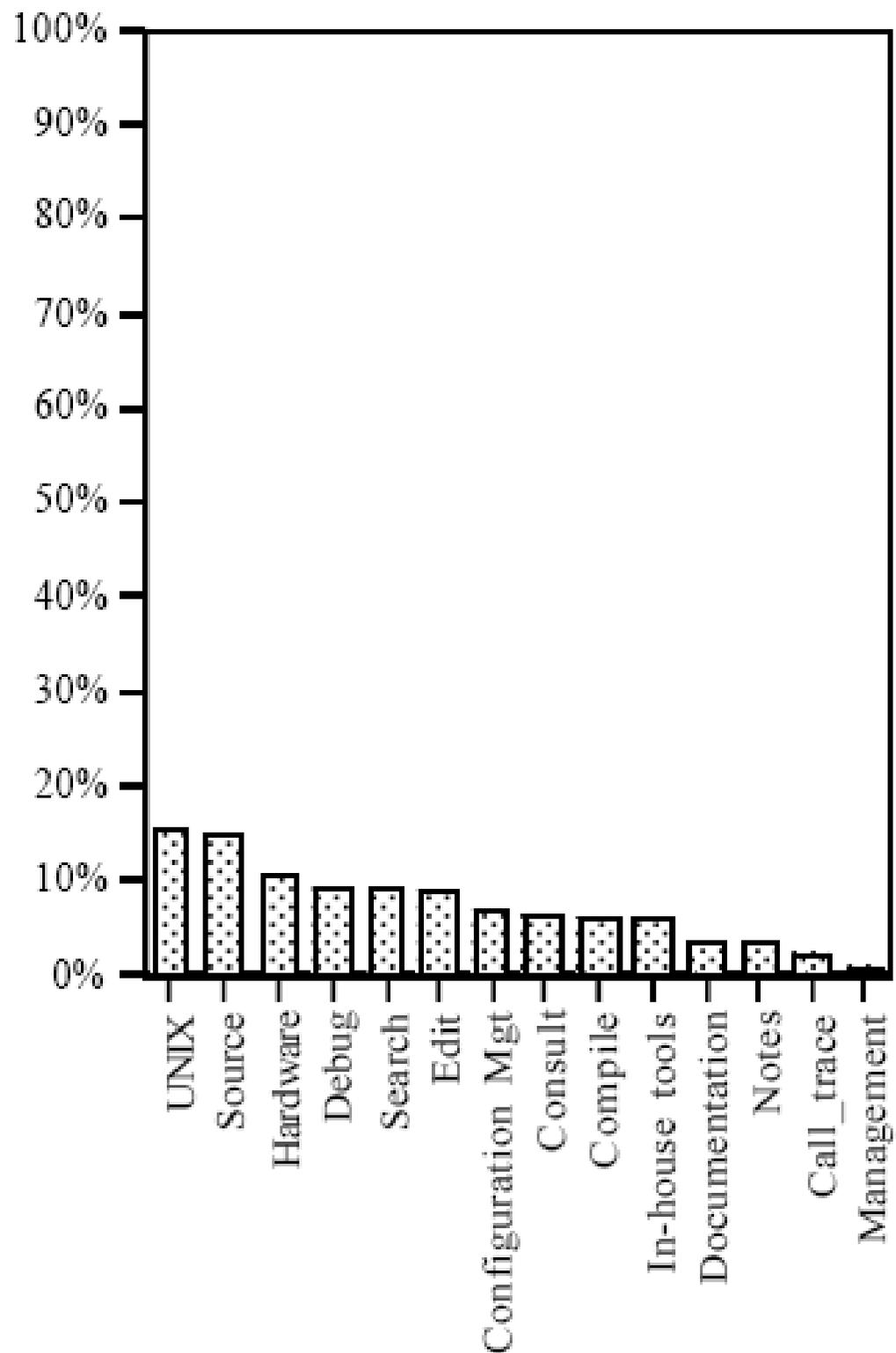


開発者観察

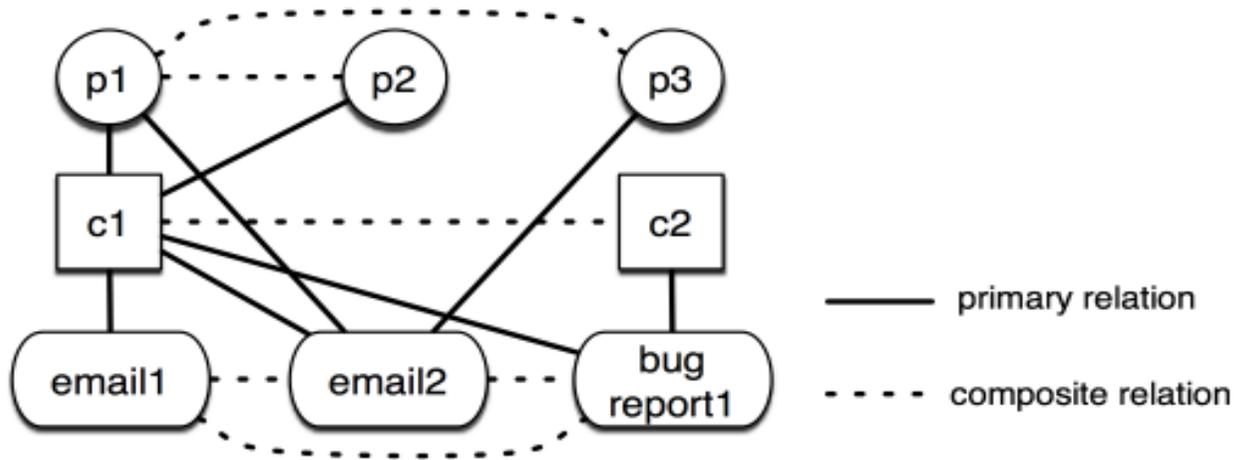
8人の開発者の開発活動を観察した結果
(合計 356 イベント)

UNIX	54回
デバッグ	52回
ハードウェア	36回
サーチ	32回
ソースコード閲覧	31回
ソースコード編集	30回

•Singer, J., Lethbridge, T., Vinson, N., and Anquetil, N., "An examination of software engineering work practices," in *Proceedings of the 1997 conference of the centre for advanced studies on collaborative research (cascon'97)*, 1997, pp. 21 pages.



ソフトウェアプロジェクトがSTISを生成する



STISのノード

p1

開発者

c1

さまざまな粒度のコードの集まり

file, class, module, block, statement

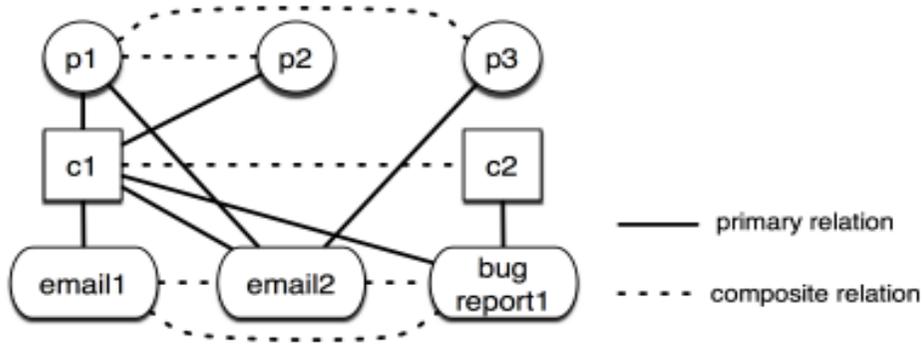
d1

各種ドキュメント

requirements, design, cvs logs, emails, bug reports

STISの関連

構造的 / 概念的 / 開発履歴的



ソーシャル依存度 SocialDependence(p_1, p_2):

$$\exists c_1, c_2 : \text{Modify}(p_1, c_1) \ \& \ \text{Modify}(p_2, c_2) \ \& \ \text{Use}(c_1, c_2)$$

ソーシャルカップリング SocialCoupling(p_1, p_2):

$$\text{COUNT}(\exists c : \text{Modify}(p_1, c, t_1) \ \& \ \text{Modify}(p_2, c, t_2) \ \& \ (t_1 - t_2) < \delta)$$

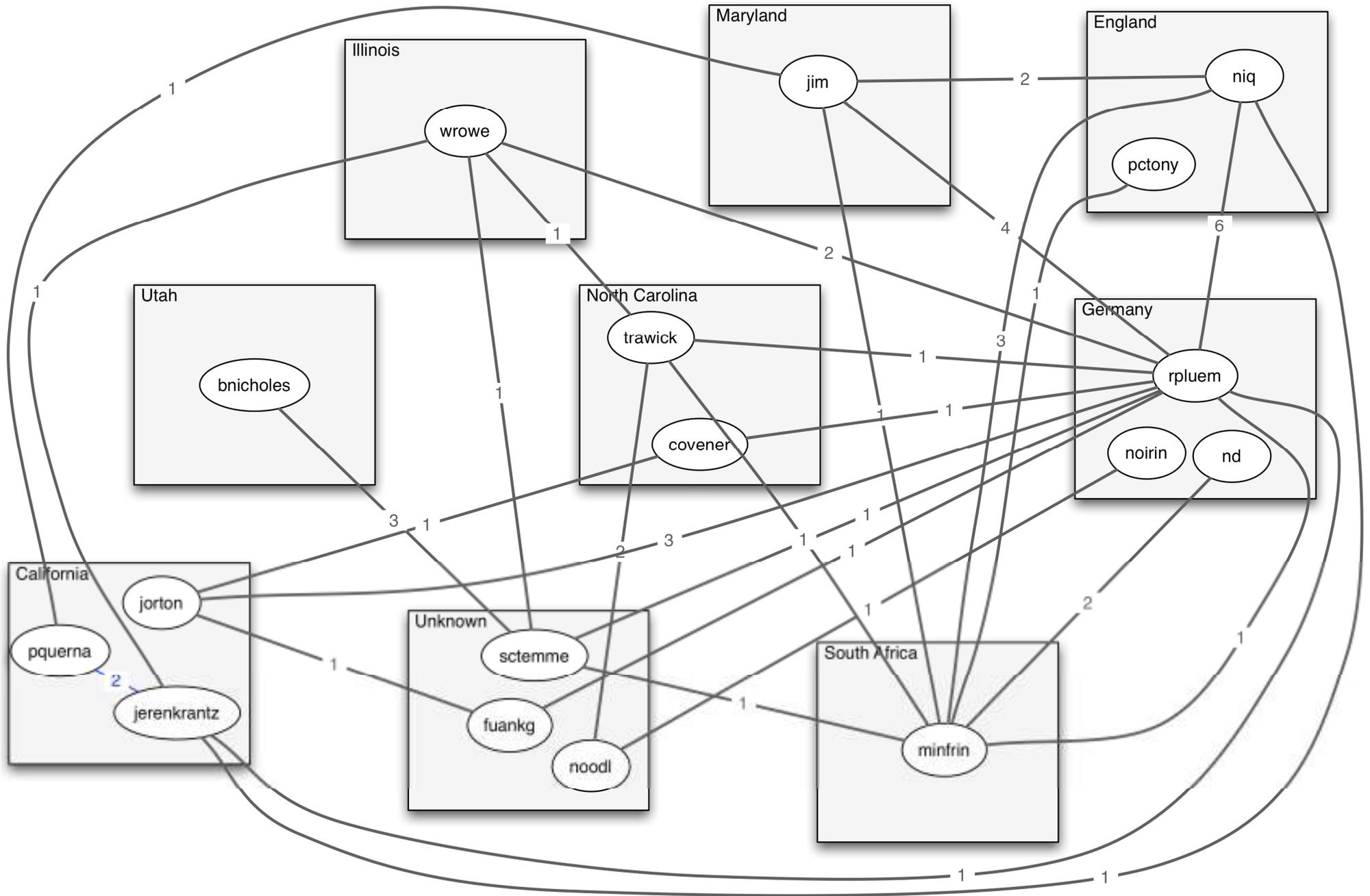
サイトカップリング SiteCoupling(s_1, s_2):

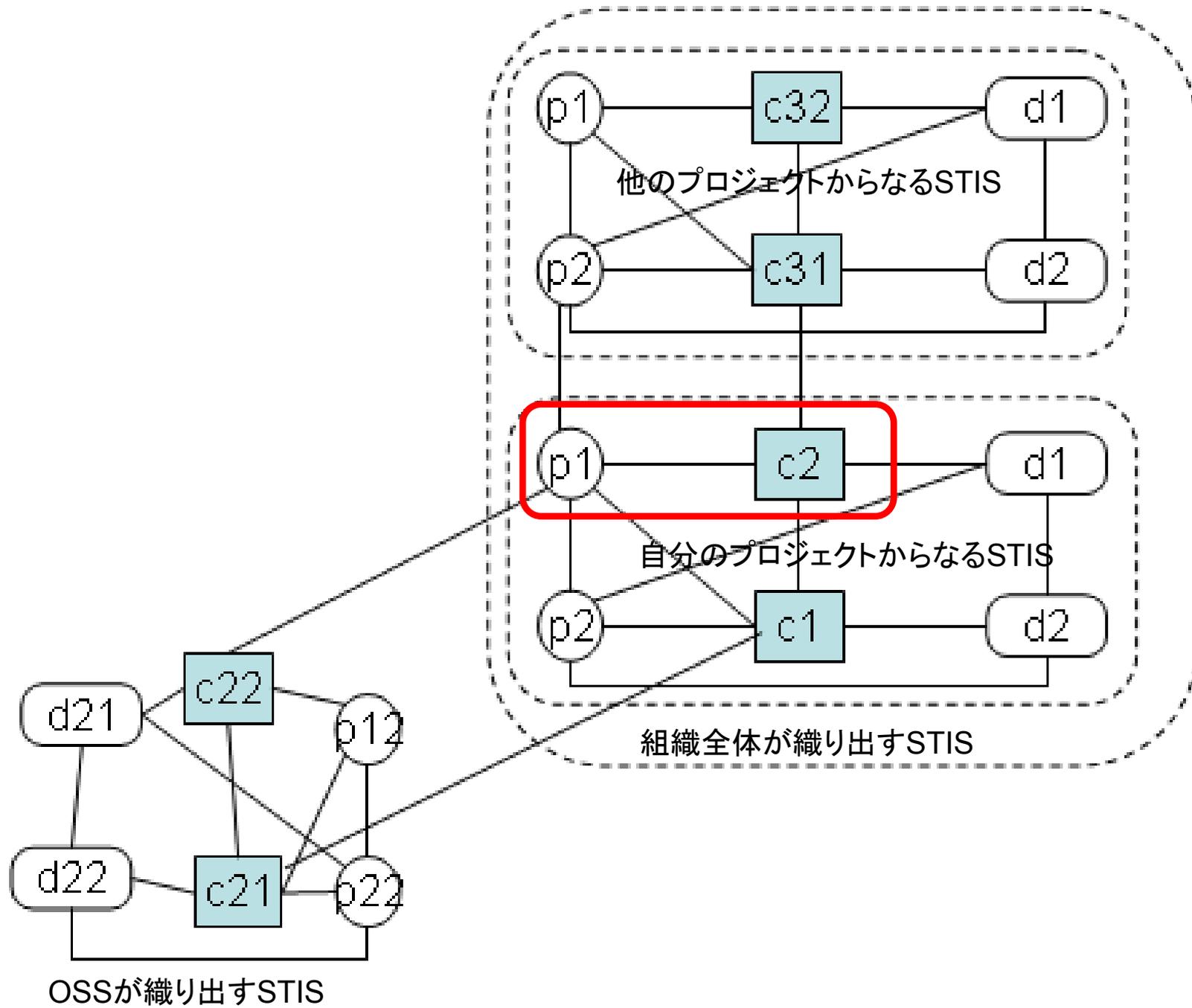
$$\text{SUM}(\forall p_m \in s_1 \ \& \ \forall p_n \in s_2 : \text{SocialCoupling}(p_m, p_n))$$

助けてくれる可能性 HelpingProbability(p_1, p_2, c):

$$\text{Modify}(p_1, c) \ \& \ \text{Affinity}(p_1, p_2)$$

サイト・カップリング site-coupling





codepot:コード検索

メソッド検索
を指定

キーワード
random string を
指定

キーワード
を含む場所を
メソッド定義に指定

入力変数は
2個のintを指定

出力タイプは
string を指定

ライセンスはASF

検索対象と検索条件を以下のフィールドに入力して検索してください。
class function both
キーワード random string を function definition から
AND を any area から 更なる検索条件
Function入出力タイプ AND int int as parameter type AND string as return type
検索の範囲 プロジェクト: パッケージ: クラス: ライセンス: ASF

SingleRecordFiller.java#SingleRecordFiller#randomString(int) in org.apache.derbyTesting.perf.clients
project: db-derby license: ASF ダウンロード
string to pick
* @return a (somewhat) random string
*/
static String randomString(int seed)

Captcha.java#Captcha#randomString(int int) in org.apache.wicket.examples.captcha
project: wicket license: ASF ダウンロード
private static String randomString(int min, int max)
{
int num = randomInt(min, max);
byte b[]

CaptchaImageResource.java#CaptchaImageResource#randomString(int int) in org.apache.wicket.extensions.markup.html.captcha
project: wicket license: ASF ダウンロード
private static String randomString(int min, int max)
{
int num = randomInt(min, max);
byte b[]

OERandom.java#OERandom#randomDecimalString(int int int) in org.apache.derbyTesting.system.oe.util
project: db-derby license: ASF ダウンロード
public String randomDecimalString(int start, int end, int scale) {

PrjInfo Note
単純検索条件例
Project: String Builder
Package: jp.co.sra.*
Project: Jun4Java721
License: ASF
ノートの付け方
• プログラム全体またはその一部分にノートをつけることができます。作成されたノートに対して、コメントや注釈などが付け加えられます。
• ログインしないと、書き込んだノートが保存されません
• ユーザ登録しない場合に、パスワードなしのGuestユーザでログインし、ノートをつけることができます。
• 左側に表示されているプログラム全体にノートをつけるには、「新規ノート」をクリックしてください。
• プログラムの一部分(一行または連続する複数行)にノートをつける場合には、マウスで対象する行を選択してください。ただし、選択する際に開始行はその行の途中から、終了行はその途中までを選択するようにしてください。
Feedback
CodeSearchについてのバグリポート、ご感想とご要望などをプロジェクトのフォーラムにお書きください。社内プライベートHPのアカウントとパスワードでログインしてください。

codepot: 知識の蓄積と共有

Results **RandomStringUtils.java** X

```
86     }
87
88     /**
89     * <p>Creates a random string whose length is the number of characters
90     * specified.</p>
91     *
92     * <p>Characters will be chosen from the set of alphabetic
93     * characters.</p>
94     *
95     * @param count the length of random string to create
96     * @return the random string
97     */
98     public static String randomAlphabetic(int count) {
99         return random(count, true, false);
100    }
101
102    /**
103    * <p>Creates a random string whose length is the number of characters
104    * specified.</p>
105    *
106    * <p>Characters will be chosen from the set of alpha-numeric
107    * characters.</p>
108    *
109    * @param count the length of random string to create
110    * @return the random string
111    */
112    public static String randomAlphanumeric(int count) {
113        return random(count, true, true);
114    }
115
116    /**
117    * <p>Creates a random string whose length is the number of characters
118    * specified.</p>
119    *
120    * <p>Characters will be chosen from the set of numeric
121    * characters.</p>
122    *
123    * @param count the length of random string to create
124    * @return the random string
125    */
126    public static String randomNumeric(int count) {
127        return random(count, false, true);
128    }
129
130    /**
131    * <p>Creates a random string whose length is the number of characters
132    * specified.</p>
133    *
```

PrjInfo Note

新規ノート

ノート表示

パスワード自動生成 112-114 (1)

guest	2009-05-27 16:17
guest	2009-05-27 16:17

パスワード自動生成に使える。本メソッドはアルファベットと数字からパスワードを生成する。

reply