Slide URL

https://vu5.sfc.keio.ac.jp/slide/

Web情報システム構成法
No.11 Webデータ

萩野　達也　(hagino@sfc.keio.ac.jp)

Web文書とWebアプリケーション

- Web文書
  - 大学，企業，組織のホームページ
  - ニュース
  - 製品情報
  - ブログ，twitter
  - Multimedia: 写真，動画

- Webアプリケーション
  - オンラインショッピング：本，物品，・・・・
  - オンラインバンク
  - オンライン予約：ホテル，列車，飛行機，映画，イベント，・・・・
  - オンラインゲーム
  - 検索エンジン：google，yahoo，bing，・・・・

キーワード検索 vs 全文検索

- キーワード検索
  - 文書ごとに，その特徴を示すキーワードを与えておく．
    - 文書のメタデータ
    - キーワードで索引を作っておく
  - 検索のときに与えられた語句に一致する文書を探し出す．
    - AND, OR, NOT検索

```
<html>
  <head>
    ...
    <meta name="keywords"
          content="慶應,大学,湘南"/>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

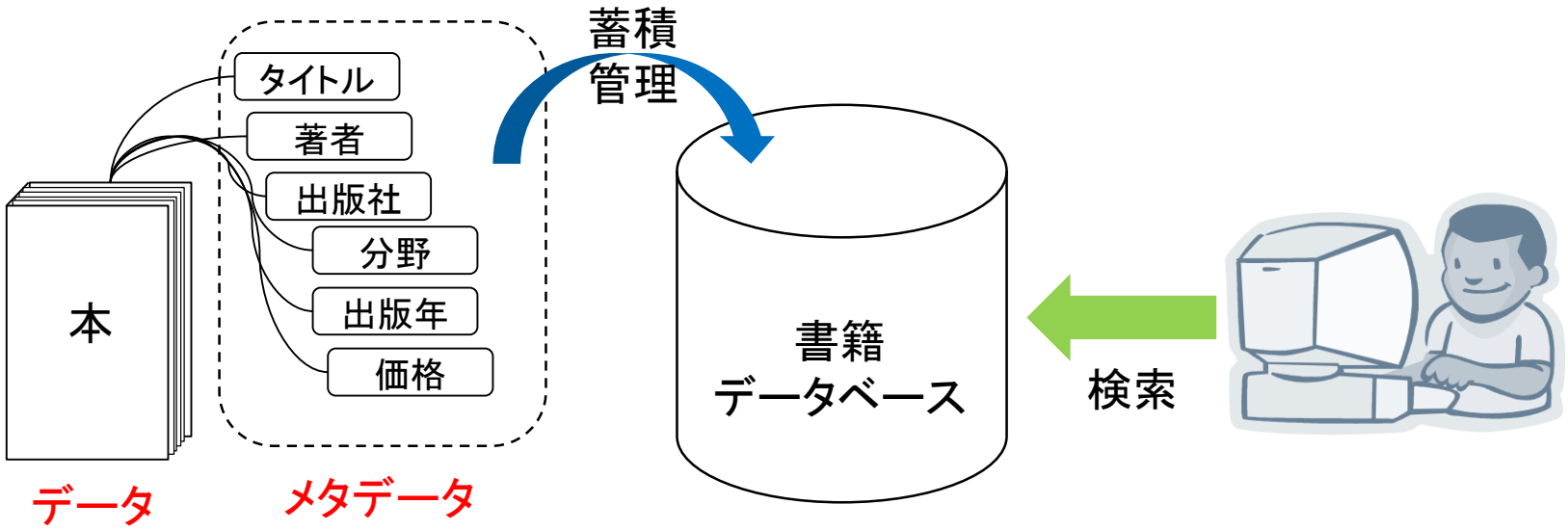
- 全文検索
  - 文書に出現するすべての語句をキーワードとする．
    - 前もって索引を作っておくべきか？
  - 検索のときに与えられた語句が文書に出現するかどうかを調べる．

	キーワード検索	全文検索
メリット	• キーワードが文書を的確に表現 • 検索が早い	• キーワードを抽出しておく必要がない • どのような語句でも検索可能
デメリット	• キーワードを前もって抽出しておく必要がある • 利用者は設定されたキーワードを知らない	• 検索結果が大量になる • 関連性の薄い文書の検索に引っかかる • 検索に時間がかかる • 索引を作成すると膨大になる

データとメタデータ

- メタデータ(meta data)
  - データについて記述したデータ

- 例:本
  - データ＝本の中身そのもの
  - メタデータ＝タイトル，著者，出版社，出版年，価格，など



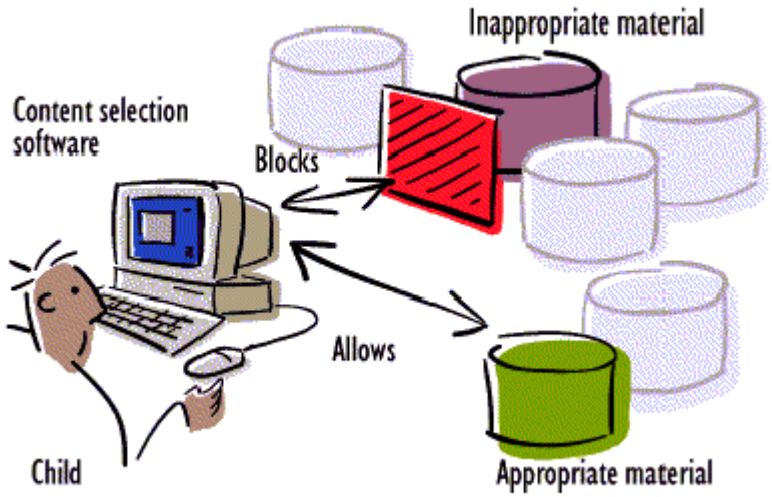
Dublin Core

- 書誌情報のメタデータ
- 15個の要素
- 30カ国(25言語)で合意

要素名	説明
Title	情報資源の名前
Creator	情報資源の内容に主たる責任を持つ者
Subject	情報資源の主題
Description	情報資源の内容の説明
Publisher	情報資源を提供している者
Contributor	情報資源の内容に貢献している者
Date	情報資源の作成日付
Type	情報資源の種類
Format	情報資源の物理的あるいは電子的形式
Identifier	情報資源を一意的に識別する文字列あるいは数字
Source	情報資源の元となった情報資源への参照
Language	情報資源の内容を記述する言語
Relation	関連する情報資源への参照
Coverage	情報資源が空間的あるいは時間的にカバーする範囲
Rights	情報資源に関する権利情報

Webのメタデータの元祖

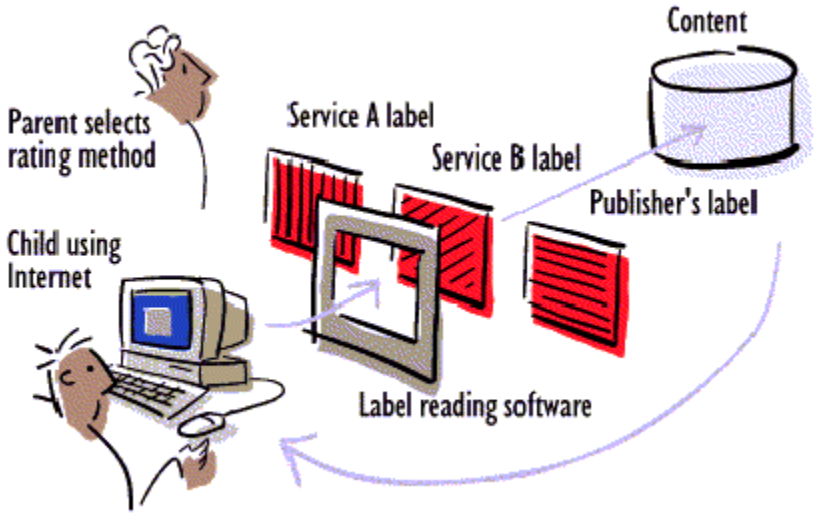
- PICS (Platform for Internet Content Selection)
  - インターネット上の有害情報へのアクセスを禁止するためのメカニズム



PICS: Internet Access Controls Without Censorship *Communications of the ACM*, 1996, vol. 39(10), pp. 87–93.

フィルタリング・ソフトウェア

- ページにつけられたラベルにしたがってアクセスしてよいかを決める



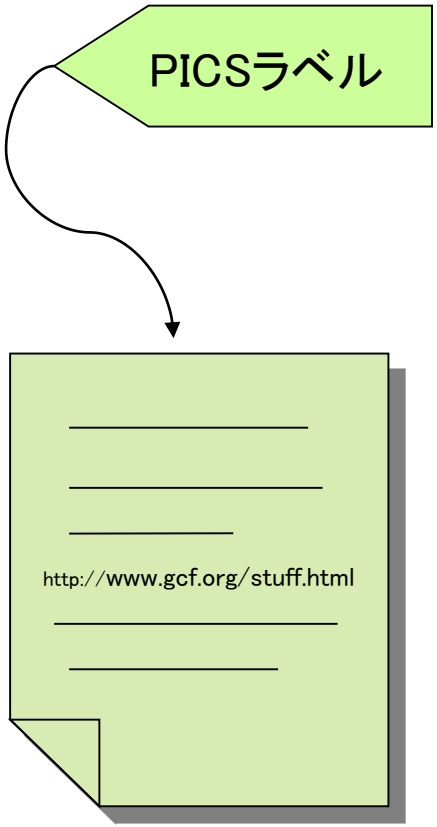
PICS: Internet Access Controls Without Censorship *Communications of the ACM*, 1996, vol. 39(10), pp. 87–93.

PICSの構成要素

- PICSレイティング・サービス
  - Rating Services and Rating Systems (and Their Machine Readable Descriptions) Version 1.1, 31 Oct 1996
- PICSラベル
  - PICS Label Distribution Label Syntax and Communication Protocols Version 1.1, 31 Oct 1996
- PICSフィルタリング規則
  - PICSRules 1.1, 29 Dec 1997

PICSラベルの例

- PICSラベルの指定方法
  - HTMLのMETAタグとして埋め込む
  - HTTPレスポンスに入れる
  - レイティング・サービスから別に受け取る




```
(PICS-1.1
"http://old.rsac.org/v1.0/" labels
 on "1994.11.05T08:15-0500"
 until "1995.12.31T23:59-0000" for
 http://www.gcf.org/stuff.html
 by "John Doe"
 ratings (l 3 s 2 v 0))
```

PICS ServiceとPICS Rule

```
((PICS-version 1.1)
 (rating-system "http://www.musac.org/")
 (rating-service "http://www.musac.org/v1.0")
 (name "Musac")
 (category (transmit-as "v")
          (name "Violins")
          (label (name "none")
                 (description "No violins")
                 (value 0))
          (label (name "Some")
                 (description "Some violins")
                 (value 1))))
```

```
(PicsRule-1.1
  (serviceinfo
    ("http://www.coolness.org/rating/V1.html" shortname "Cool" bureauURL
"http://labelbureau.coolness.org/Ratings" UseEmbedded "N")
    Policy (RejectIf "((Cool.Coolness <= 3) or (Cool.Graphics >= 3))")
    Policy (AcceptIf "otherwise")))
```

検索エンジンは万能アプリか？

- 少ないキーワードからWeb文書を探し出す
  - 2～3個のキーワードを与えることが多い

- 検索エンジンの仕組み
  1. Webサイトをクローリングして Web文書を集めてくる
     - ハイパーリンクをたどる
  2. すべての単語をキーワードだと思い索引を作る
     - 全文検索
  3. 検索で与えられたキーワードから索引を使ってWebページを探す
     - AND, OR, NOT
  4. ページランクを使って検索結果の表示順を決める
     - よりたくさんリンクされているページのほうが重要である

索引の作成

- 検索を効率よく行うために索引を前もって作っておく.

- 索引の作成
  1. それぞれの文書からキーワードを位置とともに抽出する.
  2. 転置索引(Inverted Index)にする.
  3. キーワードから集合演算などで対応する文書と位置を計算できる.

	キーワードと位置
文書1	(キーワードA,位置A1), (キーワードB,位置B1), (キーワードC,位置C1)
文書2	(キーワードA,位置A2), (キーワードC,位置C2)
文書3	(キーワードB,位置B3), (キーワードC,位置C3)

	キーワードと位置
キーワードA	(文書1,位置A1), (文書2,位置A2),
キーワードB	(文書1,位置B1), (文書3,位置B3)
キーワードC	(文書1,位置C1), (文書2,位置C2), (文書3,位置C3)

日本語の検索

- 英語の場合には単語でキーワードを切り分けることができる.
  - ″Keio University is in Tokyo.″
    → { 'Keio', 'University', 'is', 'in', 'Tokyo' }

- 日本語の場合には単語に分けるのが難しい.
  - 形態素解析を行い，分かち書きを行う.
  - 「慶應義塾大学は東京にあります．」
    → { '慶應義塾大学', 'は', '東京', 'に', 'あります' }
  - 「慶應」や「慶應義塾」では探せない.
  - 固有名詞や新語は辞書にないことも多い.

- N-gram
  - N文字ずつに分けてキーワードとする．bigram（2文字ずつに分ける）
  - 「慶應義塾大学」
    → { '慶應', '應義', '義塾', '塾大', '大学' }
  - 「慶應義塾」を検索するときは，「慶應」「應義」「義塾」を探し，その重なりから順序を決める.
  - 索引が大きくなるのが問題．Nを大きくすると緩和されるが，N文字未満のキーワードは検索できなくなる.

検索結果の表示

- 与える検索語が少ないため大量の文書がマッチする
  - 検索結果の表示の順番を工夫する必要がある

- 順位付け
  - キーワードに重みを付ける
  - 文書に重みを付ける
  - 検索履歴から関連性を計算する
  - 文書の更新日付を利用する
  - 検索者の位置情報や嗜好を考慮する

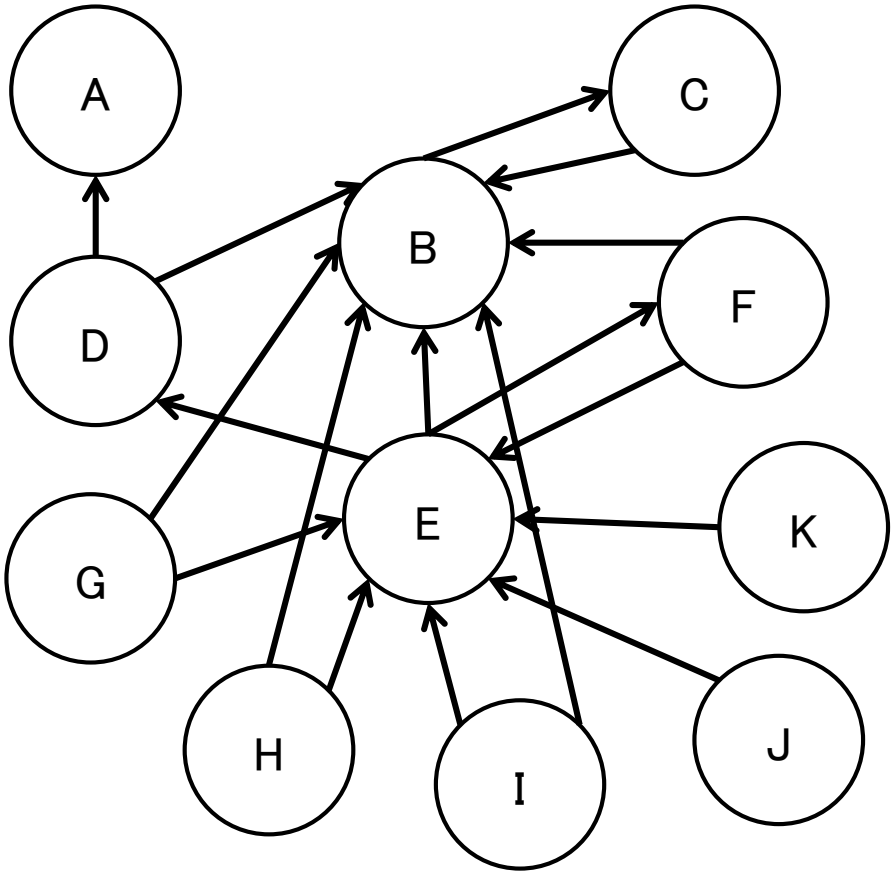
ページランクのアルゴリズム

- Webページ: $p_1, p_2, \ldots, p_N$
  - $M(p_i)$ = p_iにリンクしているページの集合
  - $L(p_j)$ = p_jから出ているリンクの数
  - N = 全ページ数
  - d = ダンピング・ファクター　0.85 (85%がリンクをたどり, 15%が別ページに直接行く)

- ページランク: $PR(p_i)$
  - $PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$

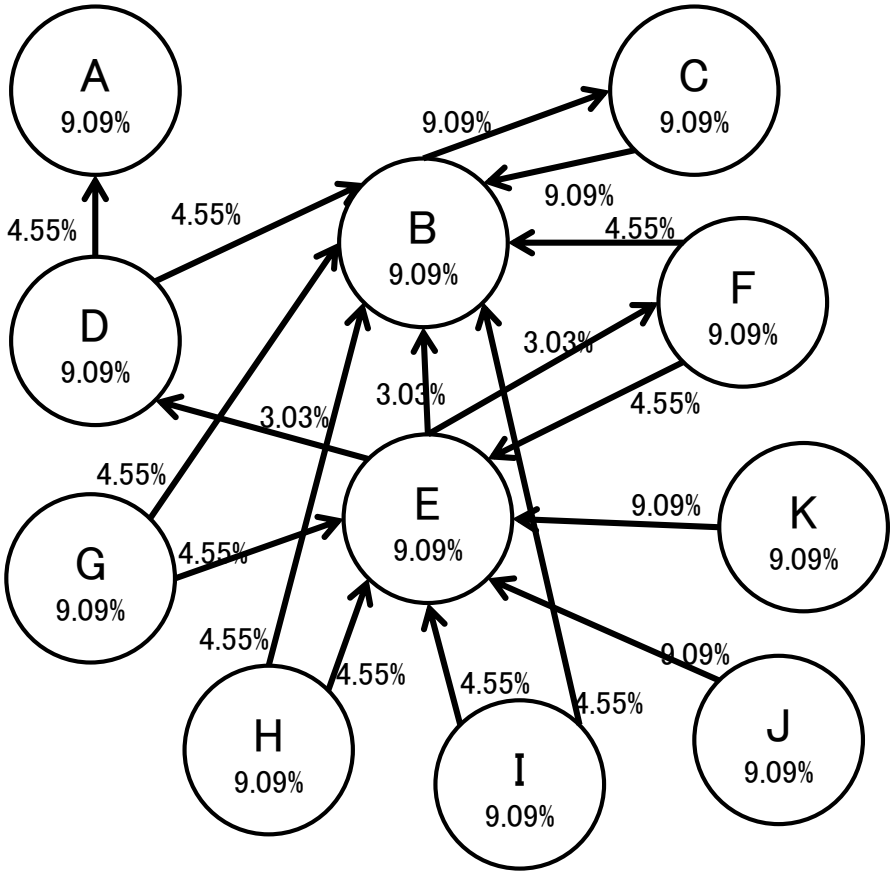
- 計算
  - ページランクの初期値: $PR(p_i) = \frac{1}{N}$
  - 上記の式を使ってページランクを更新していく

例



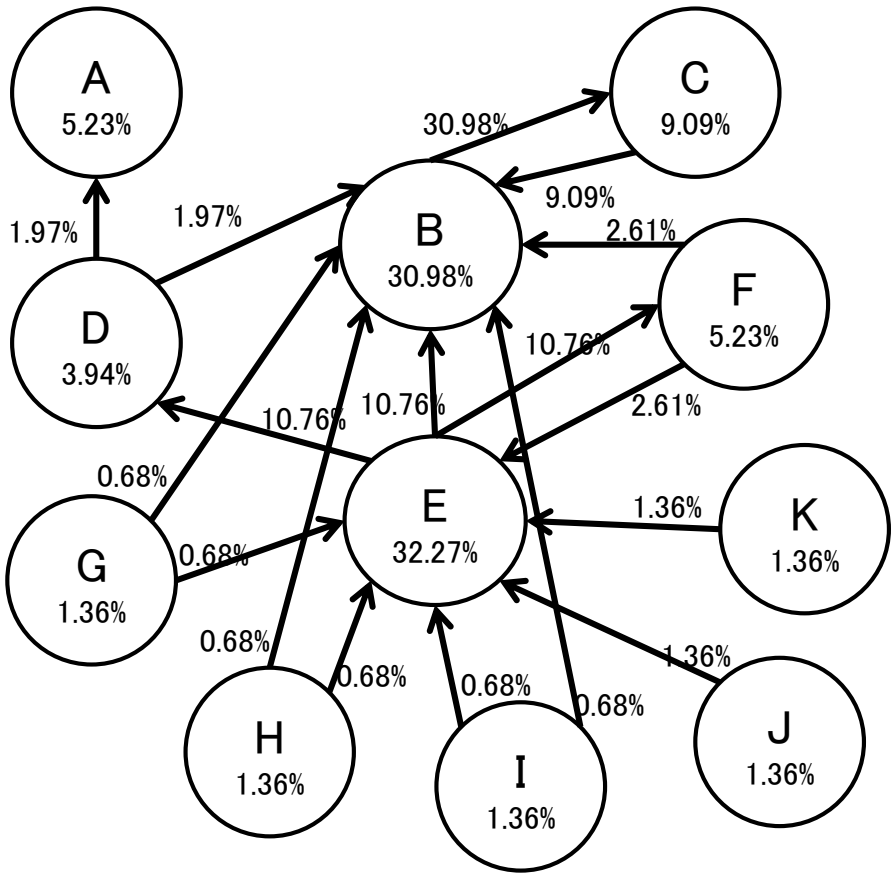
page	$M(p_i)$	$L(p_j)$
A	D	0
B	C, D, E, F, G, H, I	1
C	B	1
D	E	2
E	F, G, H, I, J, K	3
F	E	2
G		2
H		2
I		2
J		1
K		1

例（t=0）



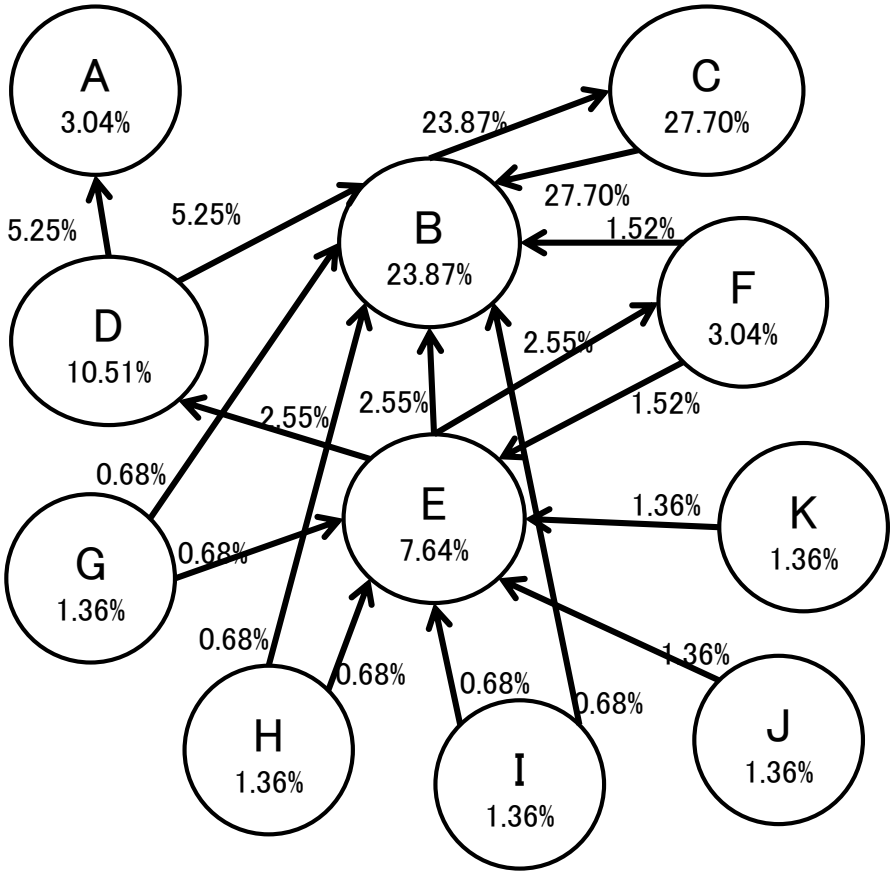
page	$PR(p_i: \mathbf{0})$	$PR(p_i: \mathbf{1})$
A	9.09%	5.23%
B	9.09%	30.98%
C	9.09%	9.09%
D	9.09%	3.94%
E	9.09%	32.27%
F	9.09%	5.23%
G	9.09%	1.36%
H	9.09%	1.36%
I	9.09%	1.36%
J	9.09%	1.36%
K	9.09%	1.36%

例（t=1）



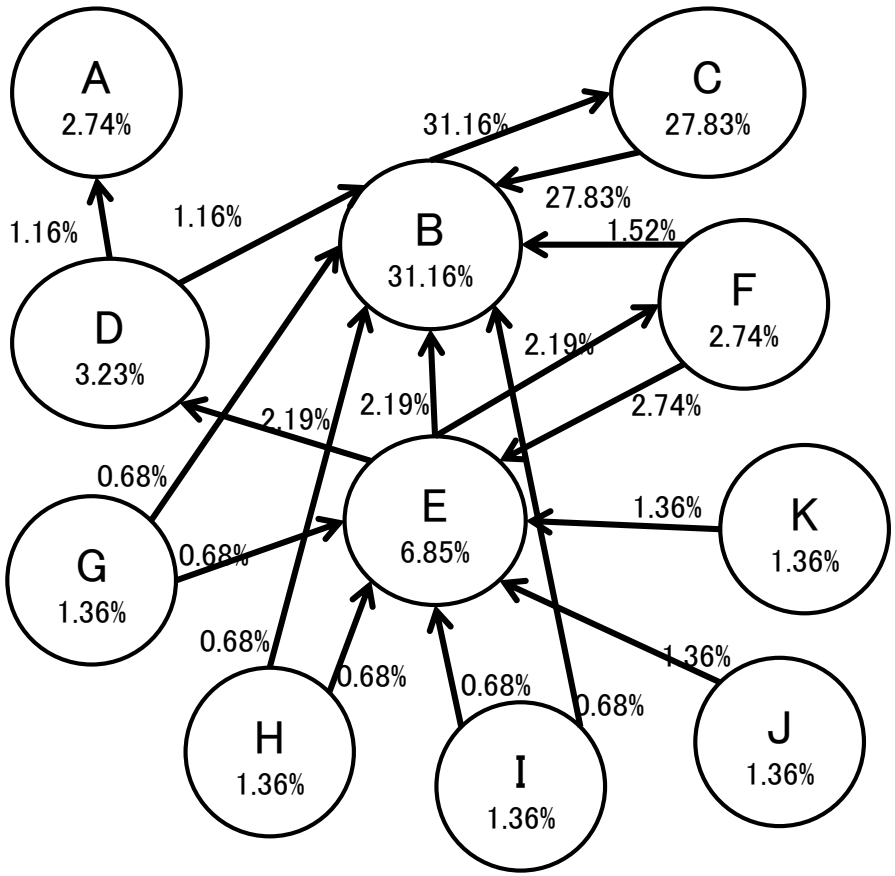
page	$PR(p_i: 1)$	$PR(p_i: 2)$
A	5.23%	3.04%
B	30.98%	23.87%
C	9.09%	27.70%
D	3.94%	10.51%
E	32.27%	7.64%
F	5.23%	3.04%
G	1.36%	1.36%
H	1.36%	1.36%
I	1.36%	1.36%
J	1.36%	1.36%
K	1.36%	1.36%

例（t=2）



page	$PR(p_i: 2)$	$PR(p_i: 3)$
A	3.04%	5.83%
B	23.87%	34.57%
C	27.70%	21.65%
D	10.51%	3.53%
E	7.64%	6.71%
F	3.04%	5.83%
G	1.36%	1.36%
H	1.36%	1.36%
I	1.36%	1.36%
J	1.36%	1.36%
K	1.36%	1.36%

例（t=30）



page	$PR(p_i: 30)$
A	2.74%
B	31.16%
C	27.83%
D	3.23%
E	6.58%
F	2.74%
G	1.36%
H	1.36%
I	1.36%
J	1.36%
K	1.36%

検索エンジンで十分か？

- 検索エンジンができること:
  - あたえられたキーワードに関連するページを探し出す
  - ページを前もって分類しておく必要はない
  - 利用者からのフィードバックも必要ない（利用しても良いが）

- 検索エンジンができないこと:
  - 隠れたページ（イントラのページやパスワードで保護）や動的に作られたページは探すことができない
  - 複数のページの情報を組み合わせることはできない
  - まとめのページを自動的には作れない
  - オンライバンクやショッピングはできない

Web上のデータ

- Webには有益なデータがたくさんある：
  - 時刻表（電車，バス，大学の時間割）
  - 天気の情報（温度，湿度，気圧，・・・・）
  - 地理情報（地図，道路，ランドマーク，店，・・・・）
  - 政府の予算・決算（税金，補助金，・・・・）
  - 本，音楽，映画などの情報（ISBN，CD，DVD，音楽家，・・・・）
  - 医療情報（薬，遺伝子，・・・・）
  - 製品情報（PC，家電，・・・・）
  - TV番組情報
  - お店の情報（スーパー，レストラン，・・・・）
  - シラバス
  - ・・・・

Webにあるデータの形式

- HTMLの表
  - 良く使われる，見やすい
  - データを取り出すのが困難（データ間の関係が分からない）

辻堂駅　東海道本線 横浜・東京方面（上り）　▶土曜・休日の時刻表を表示

時	平日
5	00 27 50
6	快高 02 11 19 快籠 27 34 40 44 47 湘ラ 52 55 58
7	快高 01 04 07 10 13 16 19 22 快籠 25 29 31 34 37 快高 40 46 49 51 54 58
8	01 04 09 12 15 19 快籠 22 27 湘ラ 30 35 43 湘ラ 51 54
9	01 10 23 33 45 快籠 50 55
10	05 15 26 38 快籠 49 59
11	快籠 14 32 50 53
12	02 14 35 44 快籠 50
13	03 15 26 32 39 快籠 50 56
14	07 15 25 38 快籠 50 54
15	04 17 32 40 快籠 50 58
16	02 16 26 29 42 快籠 50 55
17	07 14 快高 19 24 34 46 快籠 51 54
18	品 05 11 快前 22 26 30 品 35 45 快籠 50 57
19	08 20 快高 23 32 42 快籠 54 58
20	快籠 09 21 25 37 快高 42 50
21	00 14 快高 21 25 39 51
22	04 19 25 33 快籠 39 45 58
23	品 品 品 12 27 39

列車種別・列車名：　無印=普通 快=快速 湘ラ=湘南ライナー
行き先・経由：　無印=東京 籠=籠原 高=高崎 品=品川 前=前橋

```
<table>
  <tr>
    <th>時</th>
    <th colspan="19">平日</th>
  </tr>
  ...
  <tr>
    <th>6</th>
    <td>快高<br />02</td>
    <td>11</td>
    <td>19</td>
    <td>快籠<br />27</td>
    <td>34</td>
    ....
  </tr>
  ....
</table>
```

CSV

- Comma-Separated Values
  - テキスト形式
  - 列はコンマで区切られる
  - データ交換によく用いられる
  - ExcelからCSVを出力可能

World Country

```
Country,Capital,Population,Area(km2),Official Languages
Japan,Tokyo,126659683,377944,Japanese
United States,"Washington, D.C.",318133000,9826675,English
United Kingdom,London,63705000,243610,English
France,Paris,66616416,640679,French
China,Beijing,1350695000,9596961,Standard Chinese
India,New Delhi,1210193444,3287590,"Hindi,English"
...
...
```

XMLで表現

- 時刻表

```
<?xml version="1.0" encoding="Shift_JIS"?>
<timetable>
  <station name="辻堂">
    <line name="東海道" dir="上り" week="平日">
      <train at="6:02" dest="高崎" kind="快速" />
      <train at="6:11" />
      <train at="6:19" />
      <train at="6:27" dest="籠原" kind="快速" />
      ...
      <train at="6:62" kind="湘南ライナー" />
      ...
    </line>
    ...
  </station>
  ...
</timetable>
```

XMLで表現

- World Country

```
<?xml version="1.0" encoding="Shift_JIS"?>
<world>
  <country name="Japan">
    <capital>Tokyo</capital>
    <population>126659683</polulation>
    <area unit="km2">377944</area>
    <language>Japanese</language>
  </country>
  ...
  <country name="India">
    <capital>New Delhi</capital>
    <population>1210193444</polulation>
    <area unit="km2">3287590</area>
    <language>Hindi</language>
    <language>English</language>
  </country>
  ...
</world>
```

HTMLの表 vs CSV vs XML

	HTML の表	CSV	XML
利点	• HTMLに埋め込むことができる • 人が読みやすい • 行や列の連結が可能 • スタイルシートを使って装飾できる	• 単純 • たくさんのアプリがCSVを取り扱うことができる • ビッグデータにもなる	• フレキシブル • 構造化データ
欠点	• 列や行の意味がはっきりしない • 大きなテーブルは作りにくい • 機械的な処理が難しい • 行や列が結合されると処理がより難しくなる	• 一つのセルには一つのデータ • 構造がない • ヘッダ行は単純な情報のみ • 長いテキストは取り扱いづらい	• 木構造のデータのみ扱うことができる • ビッグデータを扱いにくい

データベース

- 関係データベース
  - データを表として表す
  - 依存関係から正規化する
    - 主キー
  - 関係代数
    - 制限，射影，結合
  - 検索言語
    - SQL

Country

id	name	capital	population	area
1	Japan	Tokyo	126659683	377944
2	United States	Washington, D.C.	318133000	9826675
3	India	New Delhi	1210193444	3287590
…	…	…	…	…

主キー

Language

country	language
1	Japanese
2	English
3	Hindi
3	English
…	…

複合キー

Webデータの特徴

- 開世界である
  - データベースのように閉じていない
  - だれでもが，どんなことでも記述することができる
    - Anybody can say anything about any topic.
  - 新しい事実が次々と生まれてくるかもしれない
- 複数のデータの結合
  - 異なるデータベースの表を結合したい
  - データベーススキーマが合わないかもしれない
  - 一つのものが複数の名前を持つかもしれない
- 矛盾するデータ
  - 矛盾するデータが存在する
  - すべてのデータが正しいとは限らない
  - 異常なデータは排除しないといけない

データの結合

- 個人のCD管理
  - 持っているCD
  - CDの情報
  - アーティストのイベント情報
- スケジュール管理
  - 自分の予定
  - 大学の時間割
  - シラバス
  - 先生の情報
  - 電車やバスの時刻表
- くすりの管理
  - 処方箋
  - 薬の情報（副作用など）
  - 健康状態（血圧など）
- 食事管理
  - 冷蔵庫の中の食材
  - 料理の情報
  - スーパーの情報

どのようなデータが欲しいか？

- どのようなデータが欲しいか？あればうれしいか？
  -
  -
- 現在あるデータおよびその形式は？
  -
  -
  -

課題:データを探せ

- SFCに存在するデータを10個以上探しなさい
  - (Web)　　SFCの関連Web上に存在するデータ
  - (Data)　　Webにはないが，存在しているデータ
  - (Big)　　日々生成されていて，活用されずに捨てられているビッグデータ
  - (Other)　　その他

- 提出
  - https://vu5.sfc.keio.ac.jp/kadai/
  - データを上記の4つに分類して箇条書きしなさい
  - データを管理している人・所有者などを調べなさい
  - 締め切り：7月1日正午

data.txt

Webに存在する
- ...
- ...

Webにはないが存在する
- ...
- ...

ビッグデータ
- ...
- ...

その他
- ...

まとめ

- 文書とアプリケーション
  - 静的情報
  - オンラインショッピング
- 検索エンジン
  - 全文検索
  - ページランク
- 文書とデータ
  - データ形式
  - データベース
  - データの組み合わせ
  - AAAの原則：Anybody can say anything about any topic.
  - 開世界