サーチエンジンについて

幸谷智紀<tkouya@cs.sist.ac.jp>
静岡理工科大学
理工学部情報システム学科

内容

- “The Internet”の構造
- Web(World Wide Web)の構造
- サーチエンジンとは？
- サーチエンジンの効用
- サーチエンジンの分類
- サーチエンジンの実例---Google™の場合
- 全文検索サーチエンジンの内部処理
- 全文検索サーチエンジンの実装
- まとめ

“The Internet”の構造

- Internetの歴史
- Internetの構造
- IPアドレスとFQDN

Internetの歴史(1/2)

「インターネットの発祥地はアメリカです。
(中略)
そもそもの始まりは軍事研究でした。いまから20年ほど前の1970年代は，ソ連との冷戦下で，ソ連からの核攻撃にどう対処するかが真剣に論じられていました。そして，ソ連の攻撃によって軍事拠点がいくつか破壊されても通信網を維持し，全体として命令系統が維持できるネットワーク技術の開発が必要だということになりました。この研究のためにARPANET(アーパネット)と呼ばれる実験ネットワークが作られました。これにはアメリカの主要大学や研究機関が参加して，研究開発と実験を進めました。」

中村正三郎編「インターネットを使いこなそう」岩波ジュニア新書，P.40-41

Internetの歴史(2/2)

1990年以前のインターネット
- 大学・研究機関・民間企業の研究者のためのネットワーク
- 文字ベースのアプリケーション (Mail/Telnet/FTP/Gopher/...)が主流

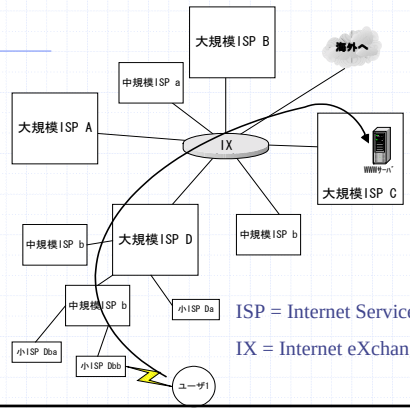
↓

1990年代初頭にWWW(World Wide Web, Web)の登場

↓

1990年代～現在まで爆発的に普及し続ける

Internetの構造



ISP = Internet Service Provider

IX = Internet eXchange

IPアドレスとFQDN

- IPアドレス = インターネット上でコンピュータ(及びインターフェース)を区別する番号

例) IPv4(32bit) : 150.38.3.5

- FQDN = ホスト名 + (サブドメイン名) + ドメイン名 =

例) www.kanto-gakuin.ac.jp
= “www” + “kanto-gakuin.ac.jp”

- DNS(Domain Name Server)がIPアドレスとFQDNの変換を担当

FQDN ← DNS → IPアドレス

Webの構造

- Webの歴史
- HTTPとHTML
- Hyper Link

Webの歴史

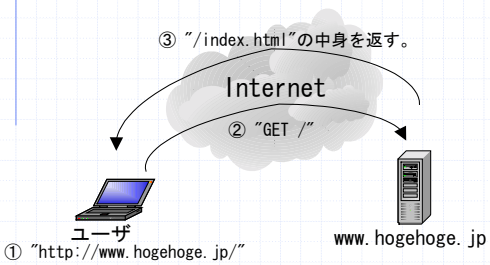
1989年・・・T.Berners-Leeが``Information Management: A Proposal“という文書をCERNに提案(この中で初めて”Web“という言葉を使用)

1993年・・・Illinois大学のNCSAでM.AndresenがX Window用のGUIベースのブラウザ``NCSA Mosaic"を開発。

1994年・・・Mosaic Communication社(後のNetscape Communication社)がNetscapeのVersion 1を開発。

1995年・・・Windows95の発売と共にMicrosoft社がInternet Explorerを開発。

Webの仕組み



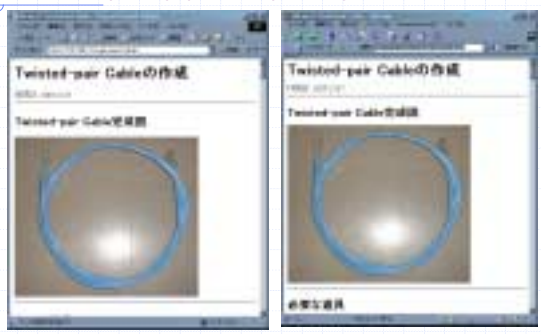
URIについて

URI = Uniform Resource Identifier
＞URL

プロトコル名　　ファイル名(絶対パス指定)
http://　www.hogehoge.jp　/index.html
サーバ名(FQDN or IPアドレス)

2強ブラウザ

Internet Explorer(IE)とNetscape Navigator



HTMLの例(1/2)

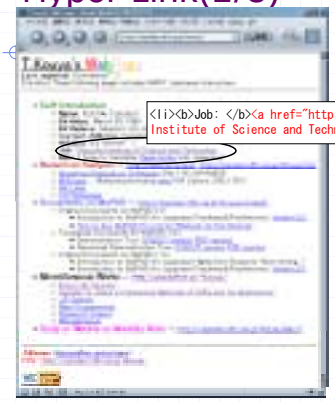
HTMLの例(2/2)

```
<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
  <meta name="Author" content="Tomonori Kouya">
  <meta name="GENERATOR" content="Mozilla/4.75 [ja] (Windows NT 5.0; U)
    [Netscape]">
  <title>My Bookmark</title>
</head>
<body text="#000000" bgcolor="#FFFFCC" link="#0000EE" vlink="#551A8B"
    alink="#FF0000">

<h1>
私のブックマーク</h1>
<img SRC="line.gif" height=18 width=640>
<blockquote><a href="http://www.kanto-gakuin.ac.jp/"><img
    SRC="kangaku.jpg" ALT="関東学院大学" height=82 width=221
    align=CENTER></a>・・・私が通っている大学です
(以下略)
```

HTMLの構造

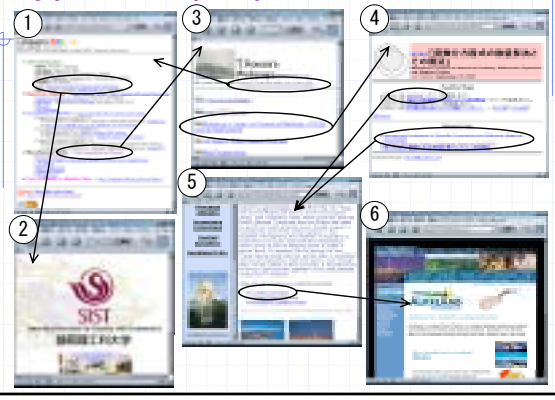
```
<HTML>
  <HEAD>
    ・・・
    <TITLE>Webページのタイトル</TITLE>
    ・・・
  </HEAD>
  <BODY>
    ・・・
    Webページ本文
    ・・・
  </BODY>
</HTML>
```

Hyper Link(1/3)

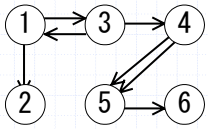


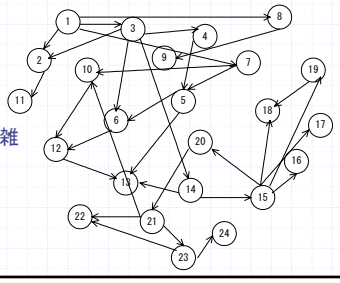
```
<li><b>Job: </b><a href="http://www.sist.ac.jp/">Shizuoka
Institute of Science and Technology</a>.</li>
```

Hyper Link(2/3)



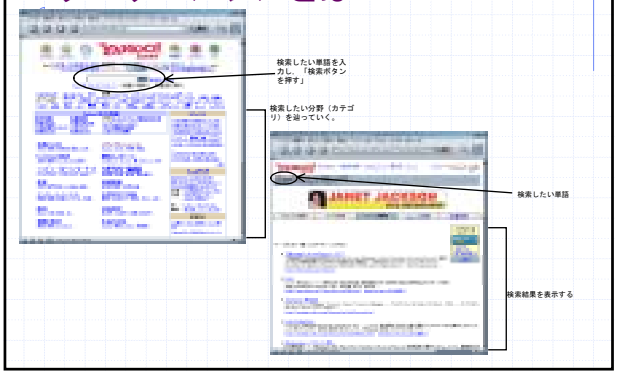
Hyper Link(3/3)
抽象化されたグラフ表現





Hyper Linkを辿っていくと複雑化していく。

サーチエンジンとは？



サーチエンジンの分類

- 全文検索型サーチエンジン
  - ・・・ユーザが見たいWebページと関連のある「キーワード」を指定することで検索するWebサイト。
- ディレクトリ型サーチエンジン
  - ・・・あらかじめWebページを，「ディレクトリ」に分類しておき，ユーザはURIをディレクトリを辿って見つける。

→現在の主な商用サーチエンジンは両方（＋α）の機能を備えている。

サーチエンジンの効用

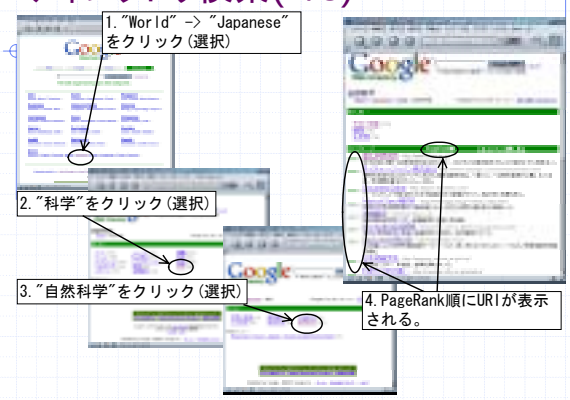
- サーチエンジンのユーザにとっては・・・
  - 未知のWebページを見つけることが出来る。
  - 自分のブラウザの「bookmark」に登録するURIを減らすことが出来る。
  - 既知のWebページが移動したり削除されたりしても探すことが出来る・・・等等
- Webページの製作者にとっては・・・
  - 広報活動の手助けをして貰える。
  - 自分のWebページのランク付けを知ることが出来る・・・等等

サーチエンジンの実例---Google™の場合

- 全文検索機能
- 画像検索
- ディレクトリ検索

- NewsGroup検索--- Usenet(News)の記事を検索

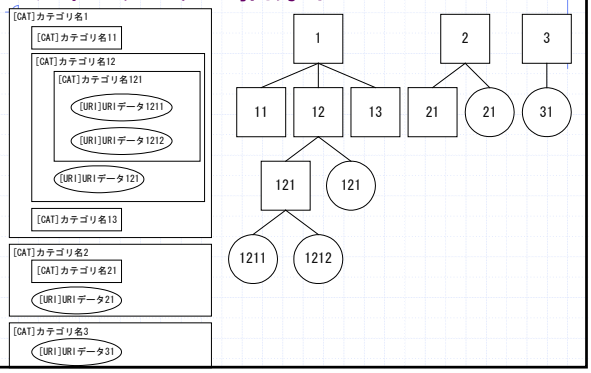
ディレクトリ検索(1/3)



ディレクトリ検索(2/3)

[CAT]World
・・・
[CAT]Italiano
[CAT]Japanese
・・・
[CAT]科学
・・・
[CAT]自然科学
[CAT]天文・宇宙
[CAT]植物
[CAT]生き物
[URI]理化学研究所
[URI]ネイチャージャパン株式会社
・・・
・・・
・・・
[CAT]Korean
・・・

ディレクトリ検索(3/3)
ディレクトリの抽象化



画像検索(1/2)

- “T.Kouya”で検索

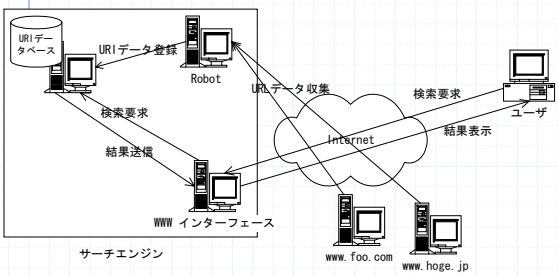
Google検索結果
→http://www.sist.ac.jp/~tkouya/
のトップページの画像を発見

画像検索(2/2)

「画像検索はどうやって動いているのですか？」

「Googleは画像，画像の説明，その他様々な要素と関連のあるテキストを解析し，画像の内容を決定しています。」
http://images.google.com/help/faq_images.html より・・・

全文検索サーチエンジンの内部処理



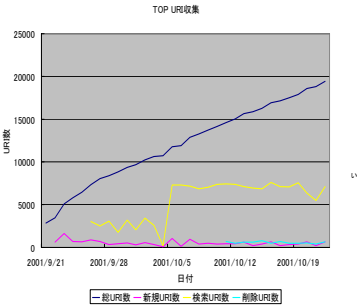
全文検索サーチエンジンの実装

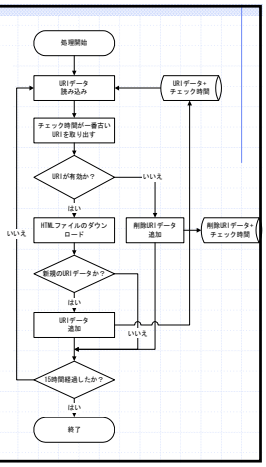
- Robot
- データベース
- 検索と結果表示
- PageRank™の思想

Robot(1/3)

- WebページのURIをHyper Linkを辿ることで発見する。
- 考慮すべきこと
  - どのようなHyper Linkを見つけるか？
  - どのようにしてWebページを探索するか？

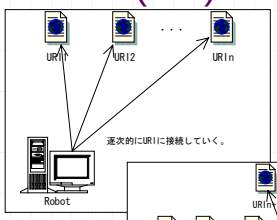
Robot(2/3)

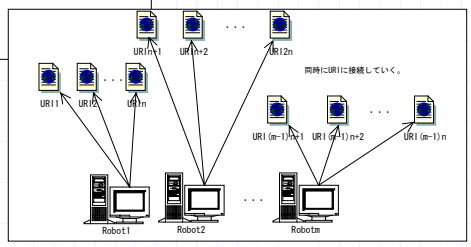




最低レベルのURI数しか収集できない。

Robot(3/3)





同じ回線を使っていても，Robotを同時に動作させることで，処理するURIを増やすことが可能になる。

例)7000URI/15時間で40Mbyte

→768Kbps*15時間*3600秒/8=約5Gbyte

データベース(1/4)

- URI
  - チェック日時
  - その他
  - キーワード1, キーワード2, ..., キーワードn
    - チェック時間の間隔
    - ディレクトリ位置1, ディレクトリ位置2, ..., ディレクトリ位置n
    - 被リンクURI1, 被リンクURI2, ..., 被リンクURI3
    - リンク先URI1, リンク先URI2, ..., リンク先URI3
    - PageRank™
    - ・・・

データベース(2/4)

- 全文検索タイプでは，関連する全てのキーワード(単語)を抽出する。
  1. HTMLを解析し，タグを取り除く→URIデータのみ別に処理する
  2. テキストから単語を抽出する。
  3. 「URI←→単語1, 単語2, ..., 単語3」という対応付けをデータベースに保存する。

データベース(3/4)

「T.Kouya's Web Page
ここは幸谷のページです。
- MuPADへのリンク
- 数値解析へのリンク
- その他のページへのリンク」

文法に基づいた解析が出来ないと，重要なキーワードを抽出できない=「日本語形態素解析」

重要な単語のみ取り出す。
固有名詞>普通名詞

“T.Kouya”, “Web Page”, “幸谷”, “MuPAD”, “リンク”, “数値解析”, “ページ”

データベース(4/4)

- Namazu(http://www.namazu.org/)を使って51285URI(FQDN=TopURI，JPドメインのみ)に対し2レベルリンクまでダウンロードし，データベースを作成する

→総データ量：約3.92Gbyte

検索と結果表示(1/3)

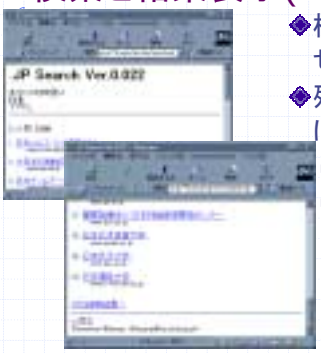
ドメインリスト

```
Registered Domains in JP (May 01 1999): 73003
Connected  Domains in JP (May 01 1999): 67304
(Domains in parentheses are not connected.)

------ JP domains:    1 (0)
 KEK              # 高エネルギー物理学研究所

------ AD domains:  215 (1)
(246              # 東京急行電鉄株式会社)
 AAA               # アーキテック・アンド・アーツ
 ABYSS              # 株式会社アビス
 ADMIRAL             # アドミラルシステム
```

検索用データ

```
www.kek.jp, "高エネルギー物理学研究所", ALIVE
www.aaa.ad.jp, "アーキテック・アンド・アーツ", ALIVE
www.abyss.ad.jp, "株式会社アビス", ALIVE
www.admiral.ad.jp, "アドミラルシステム", ALIVE
```

検索と結果表示(2/3)



- 検索部分はPerlにお任せ
- 殆どはpagingのために費やされる
  - 20URIごとに1ページ
  - 20URI以上のデータは表示済みを読み飛ばして更に検索する

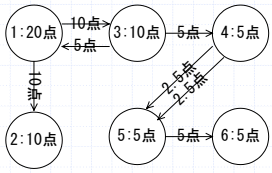
検索と結果表示(3/3)

```
#-----------------------------------#
# method = "POST" -> サーチする
#-----------------------------------#
if(&MethPost){
        # 検索処理
        exit;
}
#-----------------------------------#
# method = "GET" -> ページめくり
#-----------------------------------#
elsif(&MethGet){
        # 更なる検索
        exit;
}
#-----------------------------------#
# エラー処理
#-----------------------------------#
else{
        &CgiDie("エラーだよ");
        exit;
}
```

- GUIベースのプログラムの基本構造
  - ユーザの入力＝eventごとに実行動作を記述する
  - event =「検索ボタンを押す」「更にURLデータを表示する」...

PageRank™の思想

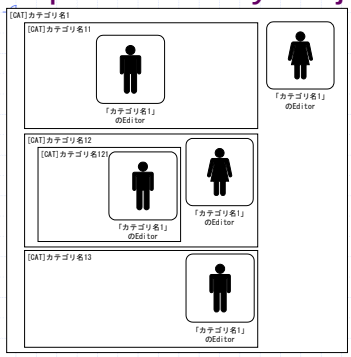
- 検索キーワード数の多さ≠求めているURI
- 被リンク数の多さ≠評価の高いURI
- 評価の高いURIからの被リンクに高い得点を与える。



サーチエンジンの今後(1/2)

- ユーザの要求に応じた選択・・・深さ優先と幅優先
- 分散型のデータ収集
  - Open Directory Project
  - 分散型サーチエンジン研究

Open Directory Project



全世界から，希望するカテゴリのEditorになれる。

サーチエンジンの今後(2/2)

- 情報の分散化→単独のサーチエンジンでは全てのWebサイトを検索することは不可能(?)→サーチエンジンの分散化
- 収集する情報を絞る＝必要な情報だけを集める