

# CloudとRich Client

---

早稲田大学  
丸山不二夫

# Agenda

---

- Server/Clientの歴史
  - Cloudの巨大化の進行
  - 無数のRich Clientの登場
-

# Server/Clientの歴史

---

# Server/Clientの歴史

---

- サーバ・クライアント・モデル('90~)
    - インターネットの登場とその爆発的な普及('95~)
  - サーバ・クライアント・モデルの変化('00~)
    - WebアプリとServer Side Programming  
汎用のクライアントとしてのWeb Browser
    - Webサービス(SOAP/REST)とSOA  
汎用のサーバとしてのWeb Server
  - Cloud時代の始まり('08~)
    - 巨大なサーバの集積
    - 無数のクライアントの存在
-

# Webアプリの全盛

---

- 人間に対してBrowserを通じてサービスを提供する。サービスの受け取り手としての人間
  - 現在のエンタープライズ・システムの中心
    - J2EE/LAMP
    - Server Side Programming
    - Multi-Tier Modelとデータベースの利用
  - Clientは、Browserの機能を使うのみ
    - Thin Clientの可能性
    - Rich Interfaceへの要求
  - AJAXの登場
-

# WebサービスとSOA

---

- サービスの担い手としてのマシン達、あるいはサービスの担い手としてのComponent達
  - マシンに対してサービスを提供する
    - SOAP: マシン上のある機能を呼び出す
    - REST: マシン上のリソースを利用する
  - SOA
    - サービスのCompositeとしてサービスを構成する
    - 業務をサービスとして切り出す
-

# サーバとクライアントのいくつかの問題

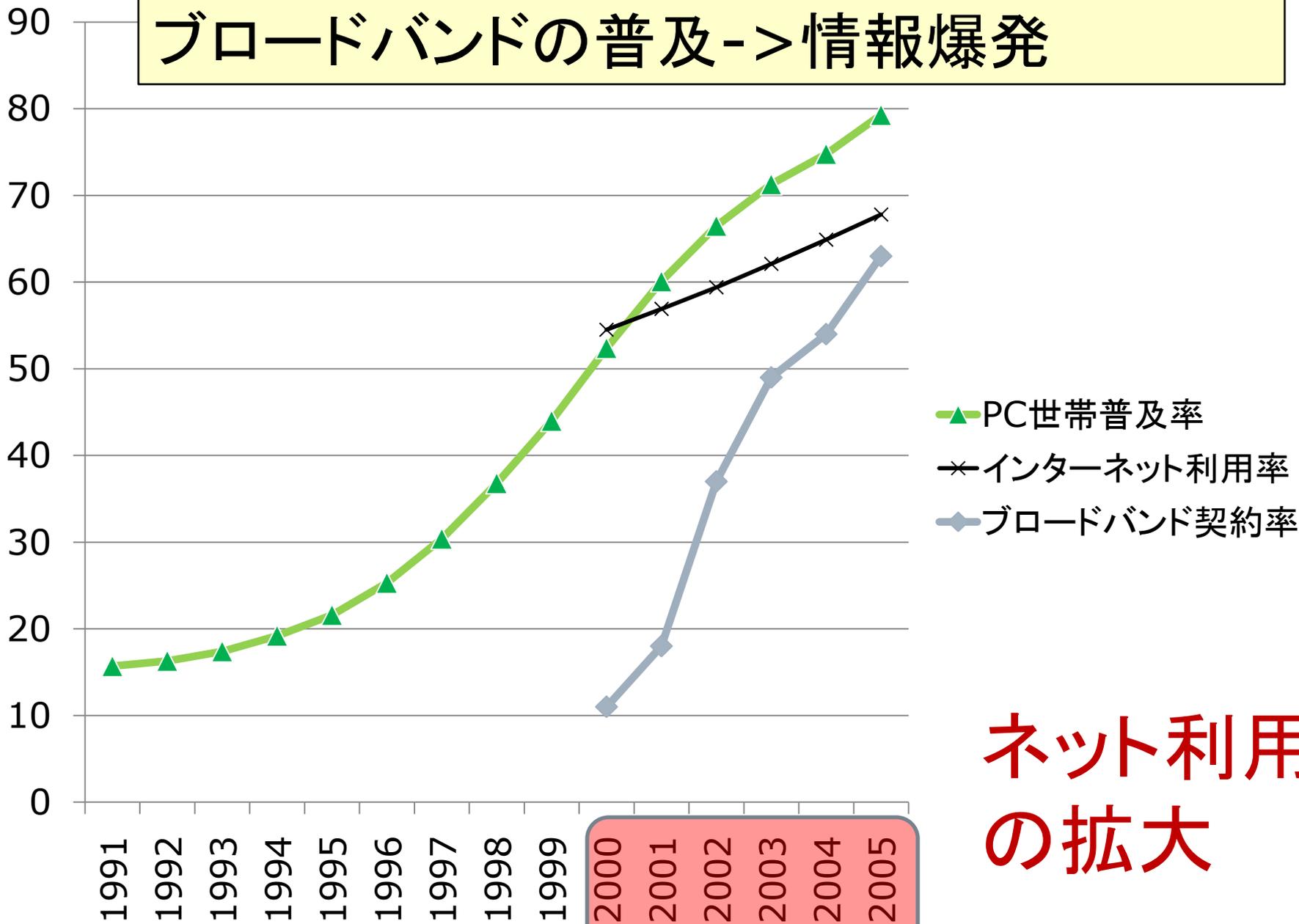
---

- Webアプリの世界では、サーバ・サイドとクライアント・サイドの陣営との分離が見られる
    - サーバ・サイド: IBM, HP, Sun, Dell ...
    - クライアント・サイド: Microsoft, Apple, Dell ...
  - クラウドのScale-out戦略は、クライアントとして使われていたコモディティ化したPCを、サーバとして利用しようというものである
    - Google, Amazon, Microsoft
  - 新しいRichクライアントとしてのMobile Deviceの登場
-

# ネット利用の拡大と Cloudの巨大化の進行

---

# PCの普及->インターネットの利用拡大-> ブロードバンドの普及->情報爆発



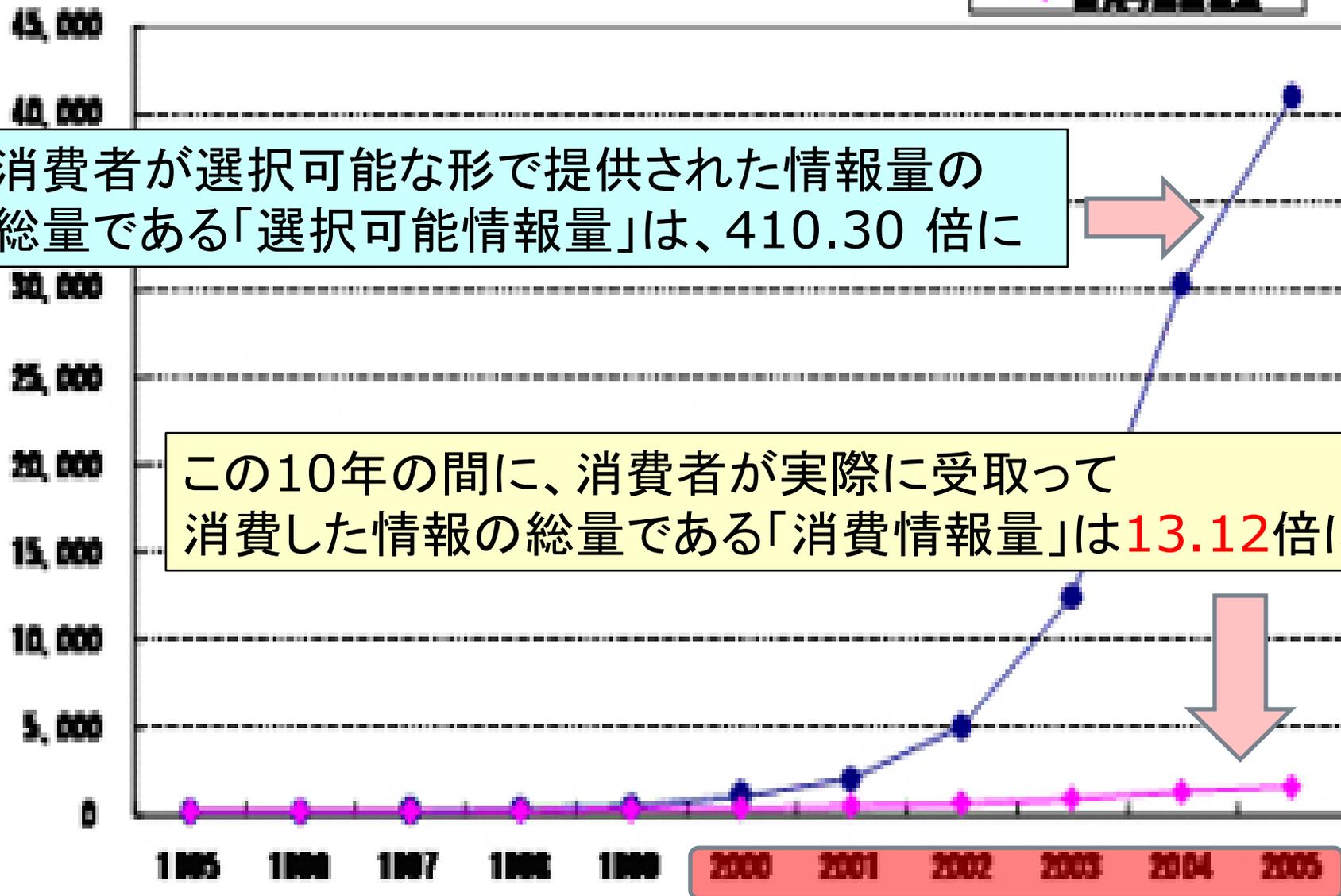
ネット利用  
の拡大

# 情報爆発



消費者が選択可能な形で提供された情報量の総量である「選択可能情報量」は、410.30 倍に

この10年の間に、消費者が実際に受取って消費した情報の総量である「消費情報量」は13.12倍に



# クラウドの巨大化の背景とその含意

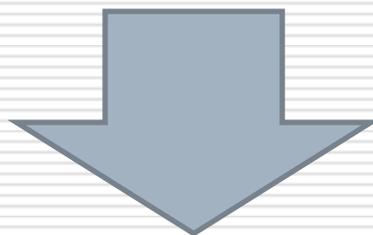
---

## □ ネット利用の拡大

膨大な数の個人が、ネットにつながり、ネットを活用しはじめたこと

## □ アプリケーションのWeb-Scaleでの大型化

## □ Consumer向けの新しいサービスの展開



□ ネット上での経済的活動・社会的行動も、こうした変化の影響を受けることは避けられない。

# Enterprise Systemの近未来

---

- Web アプリというのは、サービスの「形式」にすぎない。
  - PC上のブラウザだけを、サービスの窓口と考えるてはいないか？
  - それだけだと、Enterprise Systemは、Legacy化する危険がある。
  - ネットにつながる無数の個人を対象にした、新しい多様なサービスの可能性。
-

# 新しいサービスの成功と処理の巨大化

---

## □ 検索

- Web Pageの収集
- Indexの作成

## □ Google Page Rank

- ページのアクセス数のカウント
- あるページはどのページから参照されているか

## □ Google Ads

## □ Amazon Recommendation

- 履歴の収集と解析

## □ Mixi, GREE 足あと

---

# 無数のRich Clientの登場

---

クラウドの巨大さに対応するものは、クライアントの数の多さである。



2008年



1975年

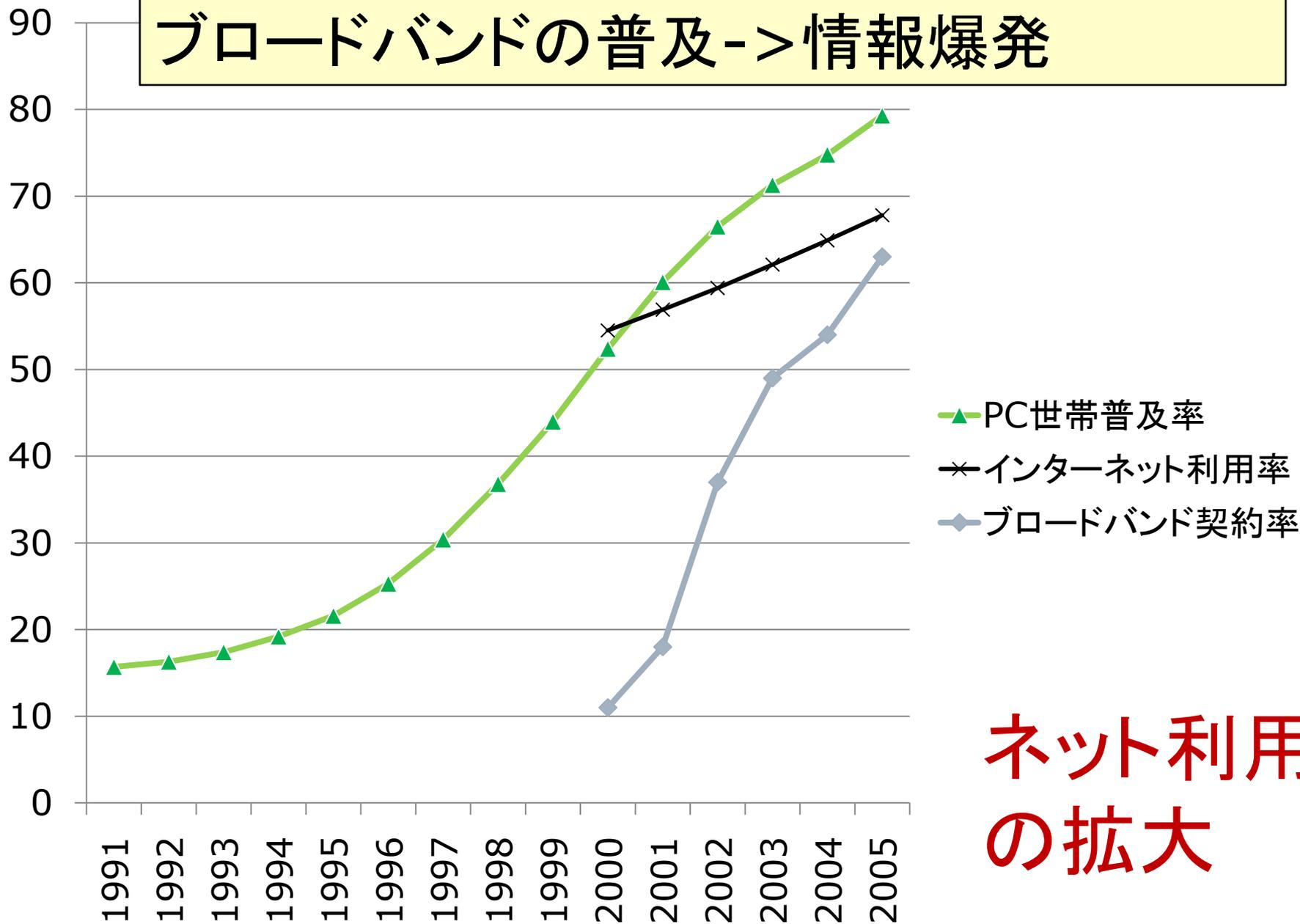
CRAY-1	Memory	4M	CPU	80MHz	500万\$
Android	Memory	192M	CPU	528MHz	178\$

# iPhone/Androidの登場

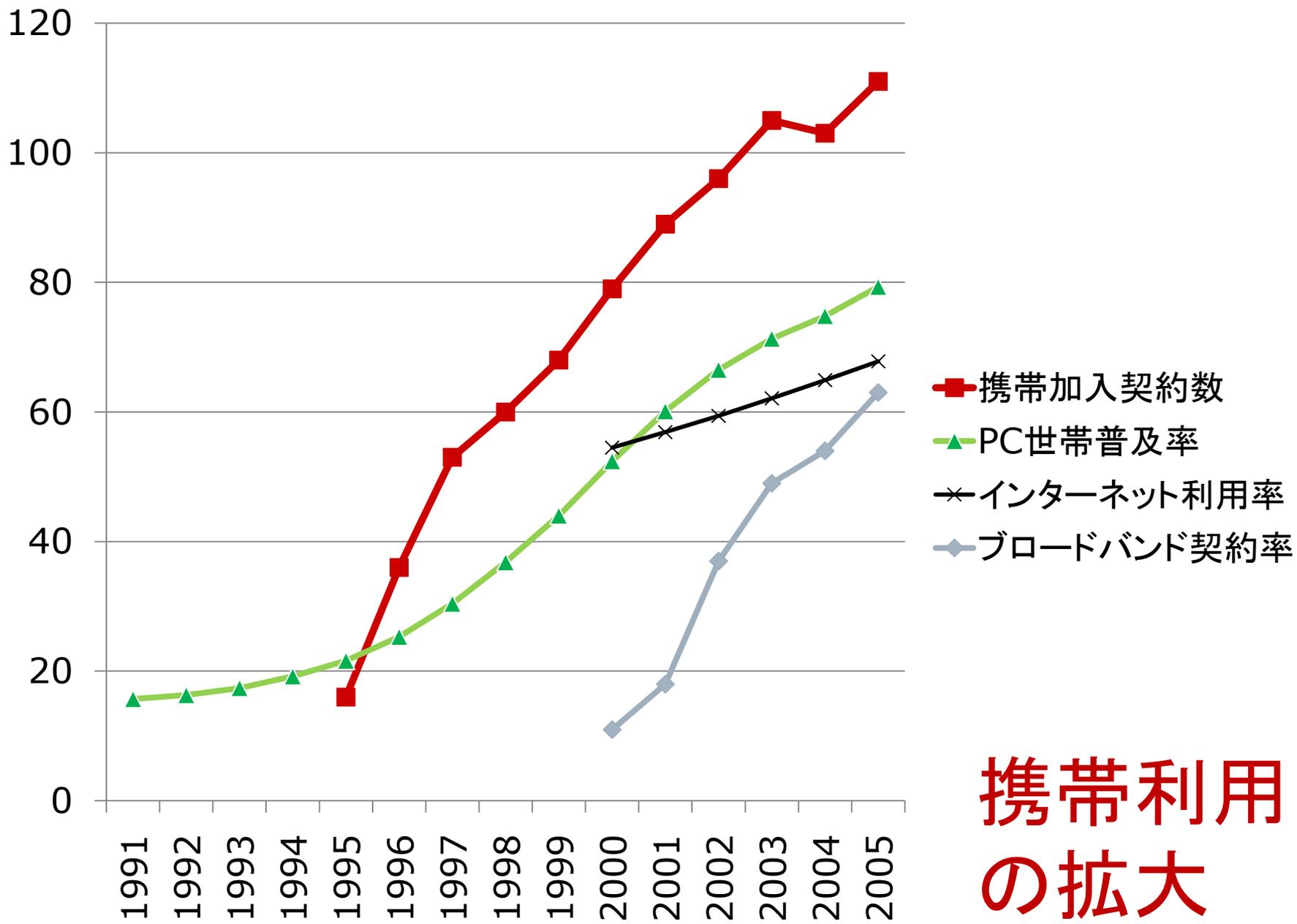
---

- 携帯の世界を、フラットにインターネットにつなぐもの。
  - カメラ、GPS、各種のセンサー、RFID、電子通貨機能の内蔵。新しい情報の産出。空間や時間へのタグ付けがもたらす、新しいサービスの発展。新しい経済活動の可能性。
  - 日本の携帯への「ガラパゴス批判」は妥当なものか？「進化の行き止まり」だったのか？
-

# PCの普及->インターネットの利用拡大-> ブロードバンドの普及->情報爆発

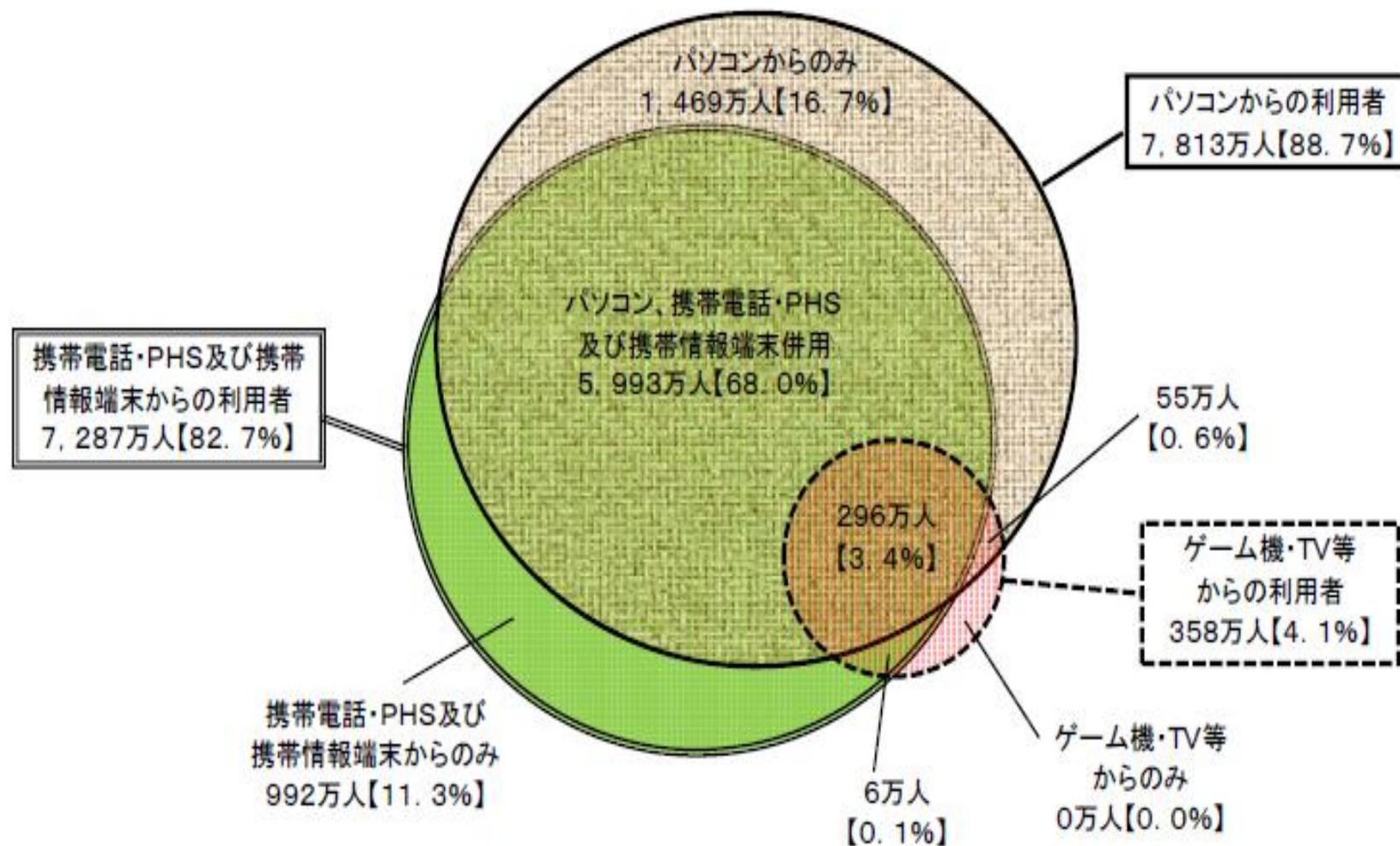


ネット利用  
の拡大



**携帯利用  
の拡大**

# 携帯でのインターネット利用の拡大



総務省「平成19年通信利用動向調査」

# インターネット利用の携帯へのシフト

---

- 個人がインターネットを利用する際に使用する端末については、携帯電話等の移動端末での利用者が平成18年末から201万人増加(対前年比2.8%増)して7,287万人となったのに対し、パソコンからの利用者は、242万人減少(同3.0%減)して7,813万人となった。
- 平成18年末から大きく変化した点としては、携帯電話等の移動端末のみでの利用者が前年から304万人増加(同44.2%増)して992万人となる一方、パソコンのみでの利用者は158万人減少(同9.7%減)1,469万人となった。

# 携帯のブロードバンド化 3.5G

---

- DoCoMo: W-CDMAを拡張した「HSDPA」を採用。通信速度は下り最大で14Mbps、平均で2～3Mbps
  - KDDI: 「CDMA 1x WIN」というサービス名で「CDMA2000 1x EV-DO」という通信方式を採用。通信速度は下り最大2.4Mbps。さらにこれを拡張した「CDMA2000 1xEV-DO Rev.A」の導入を進めている。こちらは下り最大3.1Mbps。
-

# 携帯のブロードバンド化 4G

---

- KDDI: 「UQコミュニケーションズ」WiMaxサービス。70Mbps(理論値)。CDMA(コード分割多重)ではなく、OFDM(直交周波数分割多重)と呼ばれる新しい技術をつかう。
  - Wilcom: 上り下りとも20MbpsとなるOFDMシステム
  - LTE (Long Term Evolution )  
AT&T、ベライゾン、オールテル 他
-

# 携帯のグローバルな展開

---

携帯電話について云えば、アフリカが世界でもっとも急速に成長している地域である。

2005年の、携帯の西ヨーロッパの成長率は、10%だったのに対して、サハラ以南の地域の成長率は、57%に上る。

去年2006年では、アフリカでの新規の携帯電話への加入者数は、北アメリカの加入者数を上回っている。

-- Dr.Mo Ibrahim

アフリカ最大の携帯電話会社Celtelの  
創設者・会長

(2005年) 世界には**22億**台の携帯があるが、そのうち**14億**台は、発展途上国にある。発展途上国では、携帯は、情報にアクセスするために利用されている。 -- *Leonard Waverman*

中国通信学会の劉彩・副理事長は、2010年までに中国の携帯電話加入数が**6億**件を超える見込みであることを明らかにした。

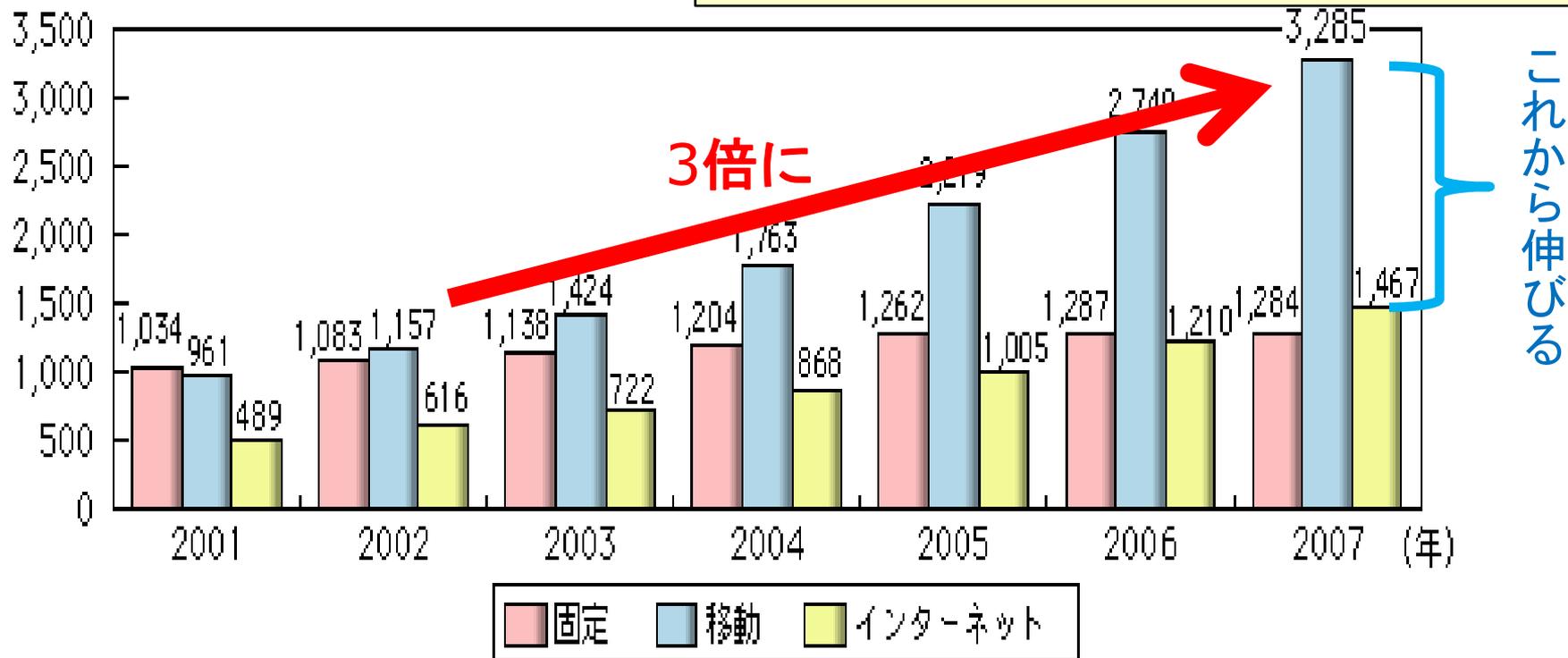
昨年8月に携帯電話台数が**2億**台を突破。インドに携帯電話が普及し始めてからアツという間であった。年内には保有台数3億台は確実。2010年には**5億**台に達するだろう。

# Mobile Deviceの新しい市場

## 携帯電話・インターネットの利用者

(電話：百万回線)  
(インターネット：百万人)

携帯電話の利用者は、この3.5年で倍増し、  
5年で3倍に増え、**32億台**に達する。



# 携帯からのインターネット接続

---

- China Mobile: TD-SDMA(中国オリジナル3G)の施設に、数兆円を投入。AndroidベースのOMS(Android + Operation Pack)を準備中。日本のi-Mode相当と見ていい。携帯からのインターネット接続が一挙に拡大する兆し。
-

## 第二の情報爆発と新しいサービス市場

---

- 携帯の爆発的な普及は、現在、なお進行中である。近い将来、50億を超えるだろう。
  - 新しい世代のモバイル・デバイスの普及とともに、携帯でのインターネット利用は、急速に進む。
  - 携帯の、ブロードバンド化は、第二の情報爆発を引き起こし、その上に新しいサービスの市場を作り出すだろう。
-

# クラウドとモバイル・デバイス

---

- 50億の携帯と500万台規模のクラウドがあったとしよう。
  - 現在のクラウド・サーバが、普通のPCであることを考えれば、単純計算では、一台のPCが、100台のクライアントの面倒を見ることになる。それは、すこし、サーバ側の能力不足である。
  - おそらく、遠くない将来、クラウドが不足することになるだろう。社会的なインフラとして、クラウドの整備は、ますます必要なものになってゆくだろう。
-

# CloudとMobile Device



# 参考資料

---

Amazon Recommendationアルゴリズム  
MapReduceの応用サンプル

# Amazon Recommendation

---

For each item in product catalog,  $I1$   
For each customer  $C$  who purchased  $I1$   
For each item  $I2$  purchased by customer  $C$   
Record that a customer purchased  $I1$  and  $I2$   
For each item  $I2$   
Compute the similarity between  $I1$  and  $I2$

<http://www.win.tue.nl/~laroyo/2L340/resources/Amazon-Recommendations.pdf>

# MapReduceアルゴリズム適用可能例

## LogからURLのアクセス頻度を累計する

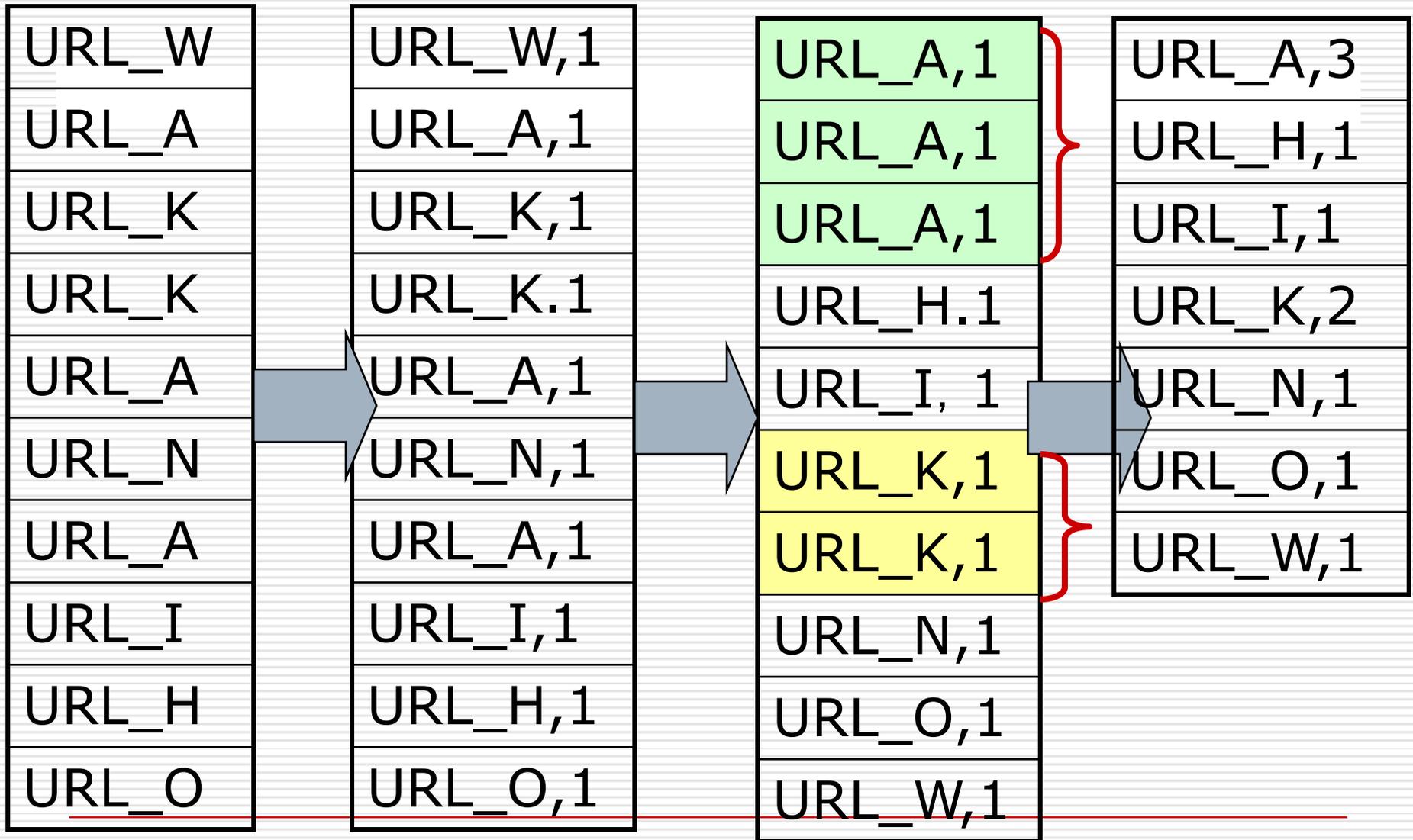
---

- mapは、Logファイルを処理して各URL毎に(`URL, 1`)を中間出力する。
  - mapreduceライブラリは、中間出力をURL毎にまとめる。(中間出力のsort)
  - reduceは、同じURLをカウントアップして、(`URL, total count`)として出力する。
-

**map**

**sort**

**reduce**



# MapReduceアルゴリズム適用可能例

## Webリンクの逆グラフ

---

- mapは、あるURLのページsourceに含まれる全てのlinkについて、そのlink先のtargetのURLを(target,source)の形で出力する。
- mapreduceのライブラリは、(target,source)の形の出力を、targetでsortする。
- reduceは、あるtargetのURLを参照している全てのsourceのリストを作って、出力する。(target,list(source))

あるページはどこから参照されているか?

# URL\_A

ref=url\_X  
 ref=url\_Y  
 ref=url\_X  
 ref=url\_B

# URL\_B

ref=url\_C  
 ref=url\_A  
 ref=url\_X  
 ref=url\_X  
 ref=url\_C

# URL\_C

ref=url\_A  
 ref=url\_X  
 ref=url\_B  
 ref=url\_Y

**map**

**sort**

**reduce**

url_X	url_A
url_Y	url_A
url_X	url_A
url_B	url_A
url_C	url_B
url_A	url_B
url_X	url_B
url_X	url_B
url_C	url_B
url_A	url_C
url_X	url_C
url_B	url_C
url_Y	url_C

url_A	url_B
url_A	url_C
url_B	url_A
url_B	url_C
url_C	url_B
url_C	url_B
url_X	url_A
url_X	url_A
url_X	url_B
url_X	url_B
url_Y	url_A
url_Y	url_C

url_A	url_B
	url_C
url_B	url_A
	url_C
url_C	url_B
	url_B
url_X	url_A
	url_A
	url_B
	url_B
url_Y	url_A
	url_C

# MapReduceアルゴリズム適用可能例

## Host毎のTerm-Vector

---

- mapは、ドキュメント毎に、そのドキュメント内での語 $w_1, w_2, \dots, w_n$ とそのドキュメント内での出現数 $q_1, q_2, \dots, q_n$ のペアのリストを、 $q_i$ の大きな順に出力する。

(hostnameA ,

(( $w_1, q_1$ ), ( $w_2, q_2$ ), ..., ( $w_n, q_n$ ))

- reduceは、これらの出力をホスト毎に集約する。  
。(hostnameA ,

(( $v_1, p_1$ ), ( $v_2, p_2$ ), ..., ( $v_n, p_n$ ))

---

# MapReduceアルゴリズム適用可能例

## 逆インデックス

---

- mapは、ドキュメントを読み込んで、一語ごとに、(word,documentID)のペアを出力する。
    -
  - mapreduceのライブラリは、これらの出力をsortする。
  - reduceは、(word, list(document ID))を出力する。
-