2019. 5. 14. 고급파일처리론 발표

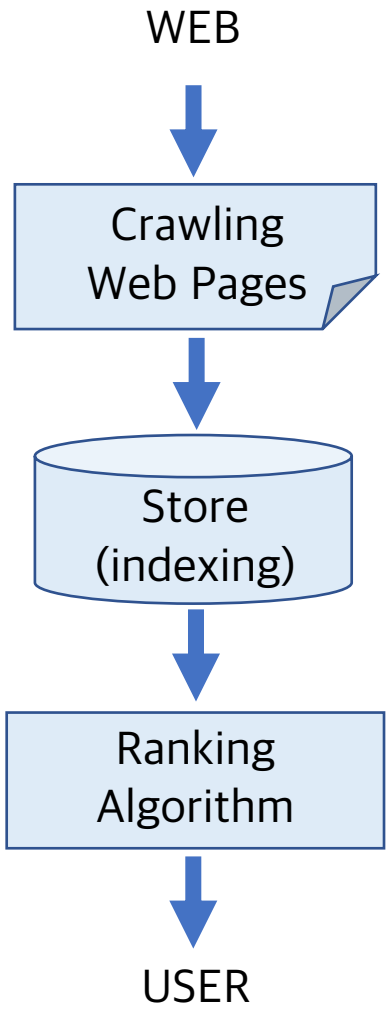
ch10. Web Search

정현진

전자전기컴퓨터공학과 병렬소프트웨어설계연구실

Contents

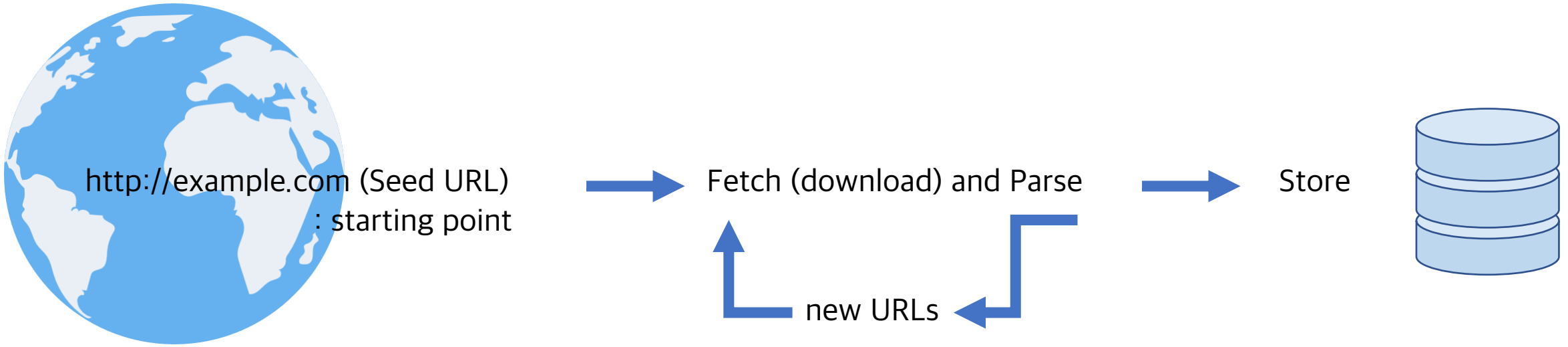
1. Web Crawling
2. Web Indexing
   - Google File System (store)
   - Map Reduce (compute)
3. Link Analysis
   - PageRank
   - HITS
4. Learning to Rank
   - Logistic Regression



1. Web Crawling

- **Crawling**

  웹 상의 페이지들을 끌어모으는 과정. (Crawler, Spider or Robot)



1. Web Crawling

- **robots.txt**

Crawler(robot)가 웹 사이트에 접근하는 것을 제한한다.

robots.txt

```
User-agent: *
Disallow: /folder/
Disallow: /file.html
Disallow: /image.png
```

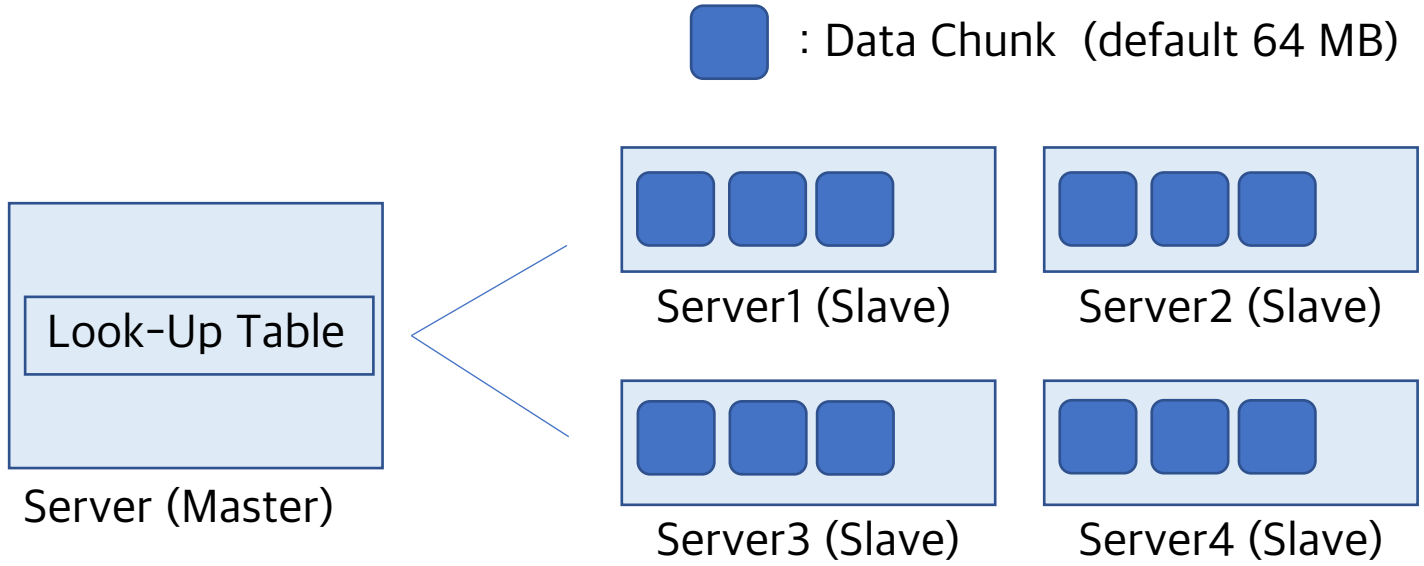
구글: Googlebot 네이버: Yeti 다음: Daumora (*: 모든 로봇)

2. Web Indexing

- **GFS (Google File System)**

  Crawler가 모은 방대한 양의 데이터를, 여러 대의 서버에 걸쳐 **분산 저장**하는 기술

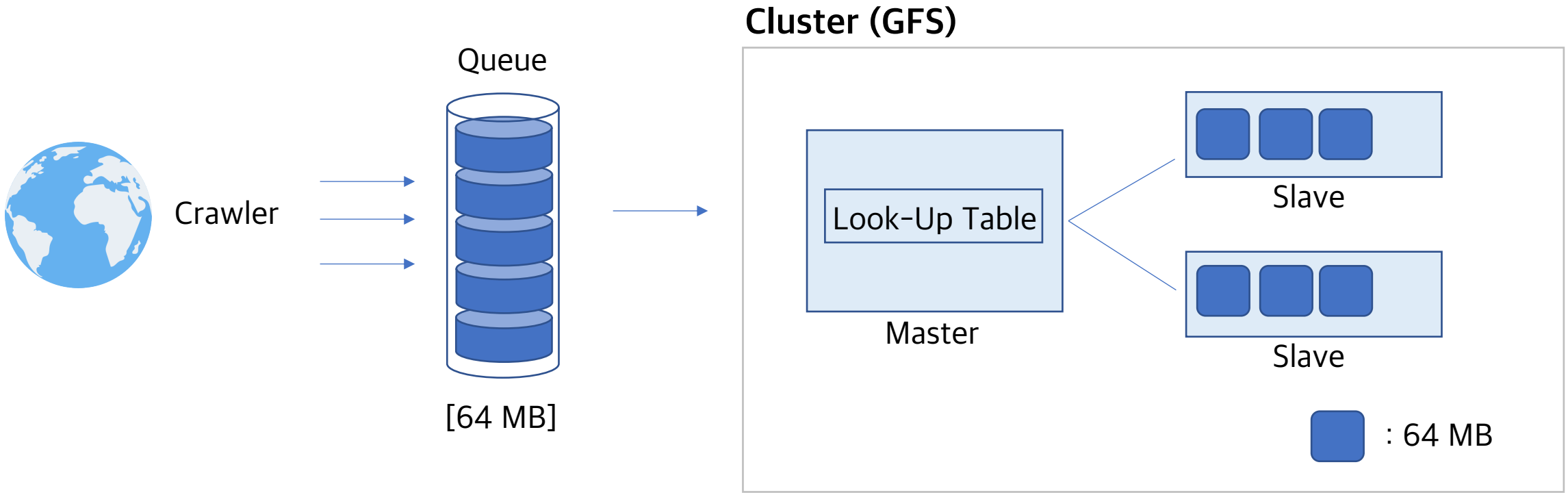
  하둡 분산파일시스템(HDFS)의 기초가 되는 기술



2. Web Indexing

- **GFS (Google File System)**

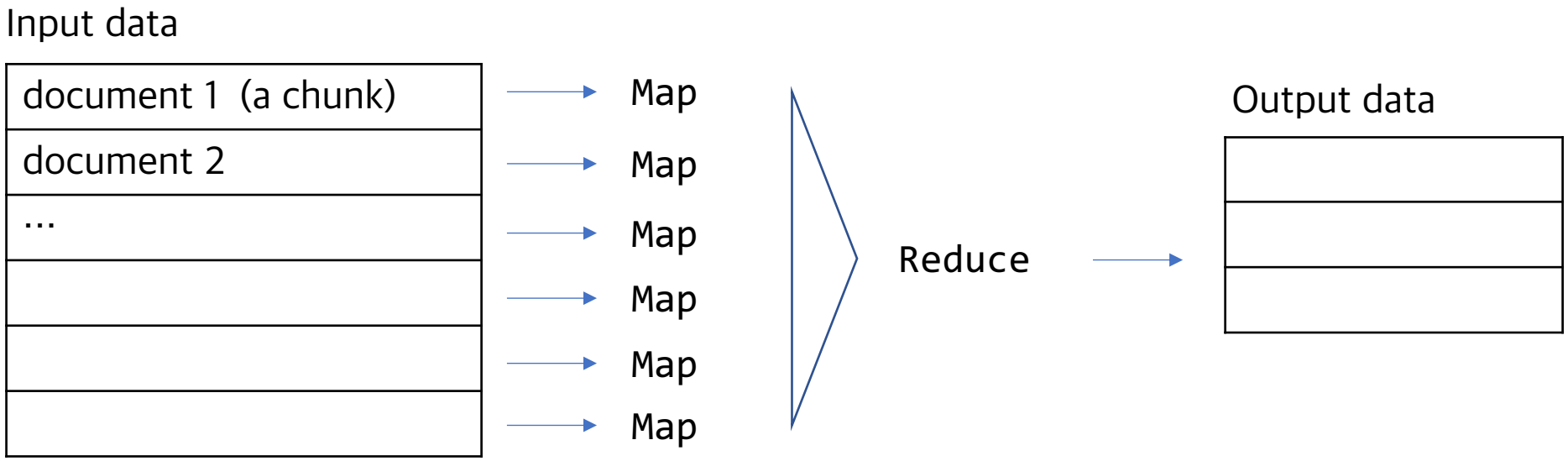
  데이터를 모아, 고정된 크기의 chunk로 만들어 저장



2. Web Indexing

- **MapReduce**

  대용량 데이터를 분산 서버에서 병렬처리하기 위해 만들어진 기술

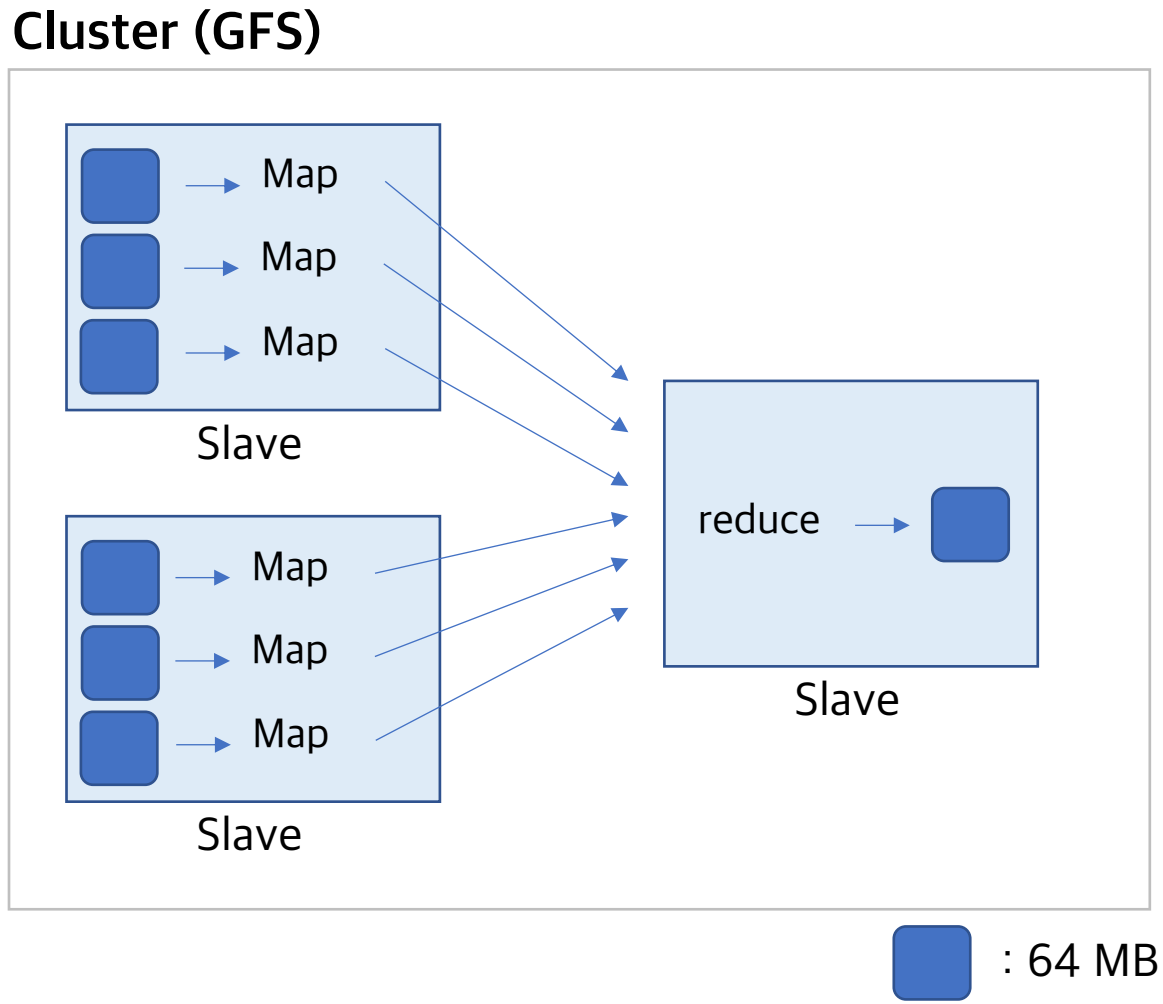


* 모든 map 처리는 독립적이어서, map들을 병렬적으로 수행 가능

2. Web Indexing

- **MapReduce**

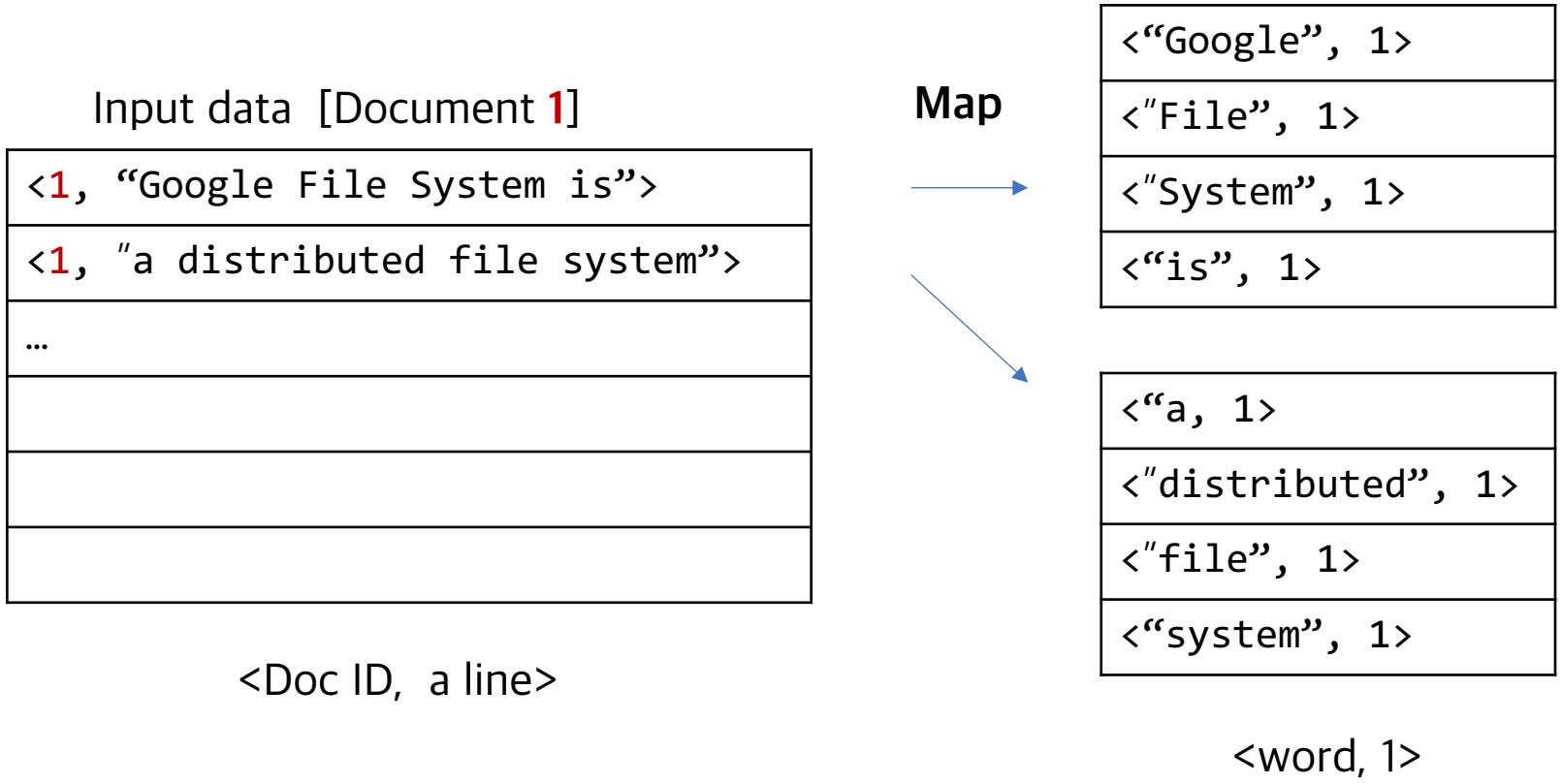
  GFS상에서 병렬 처리



2. Web Indexing

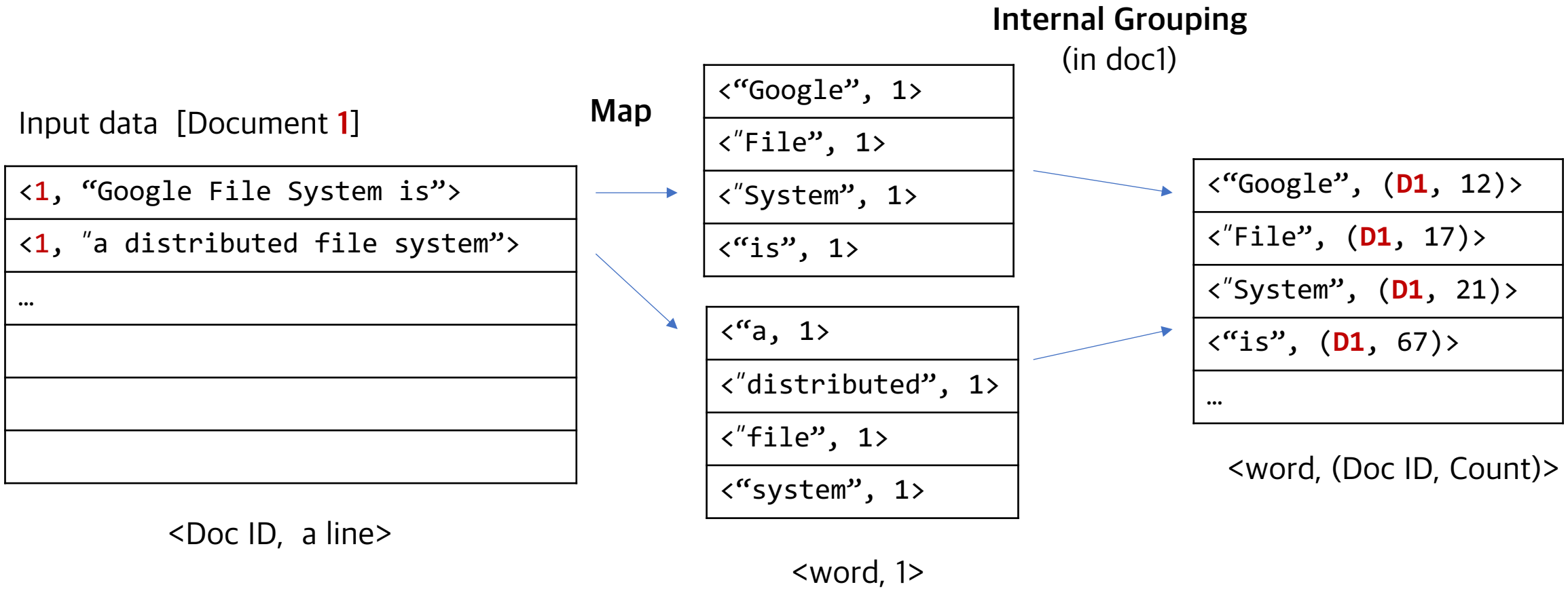
- **MapReduce**

  map, reduce의 입력과 출력은 <Key, Value> 쌍



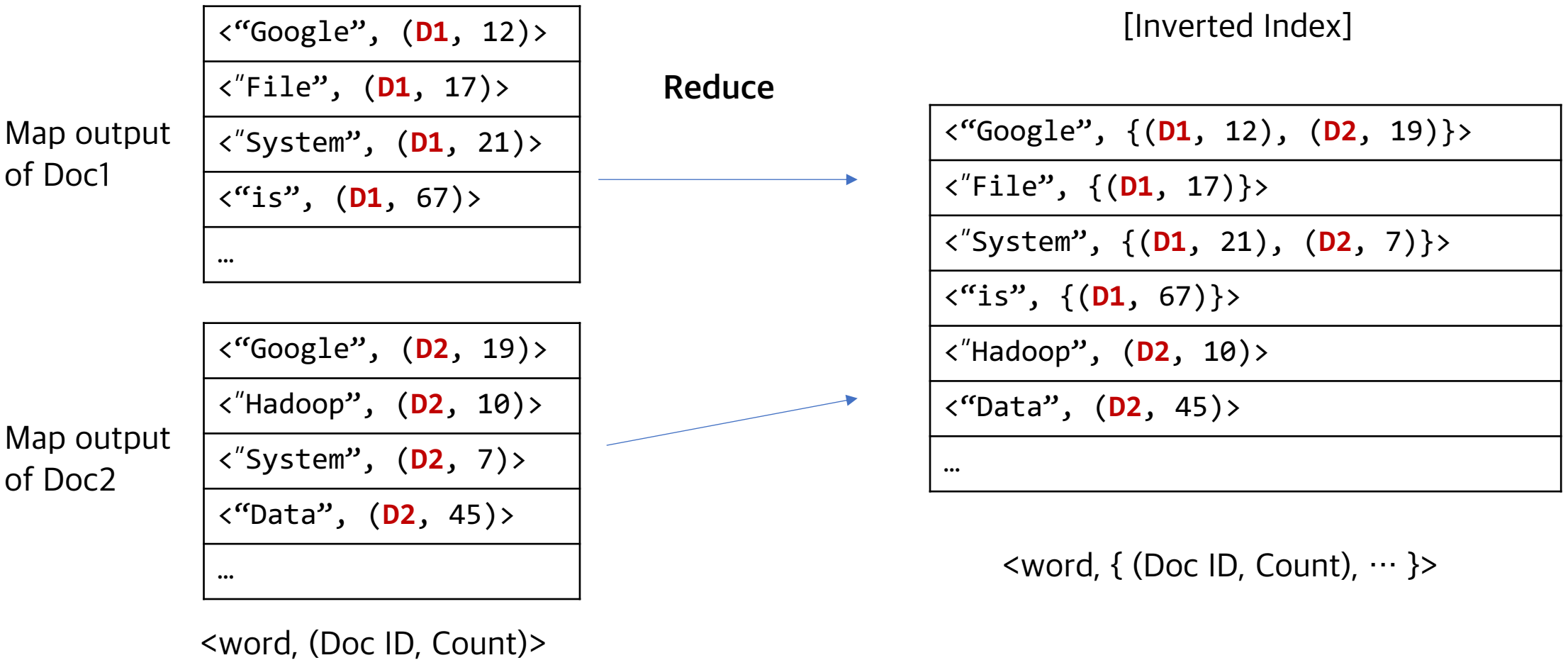
2. Web Indexing

- **MapReduce Example: Inverted Index**



2. Web Indexing

- **MapReduce Example: Inverted Index**



Ranking Algorithm

웹 정보를 검색할 때, 단순히 단어를 찾는 것 만으로는 원하는 정보를 바로 알기 어려움

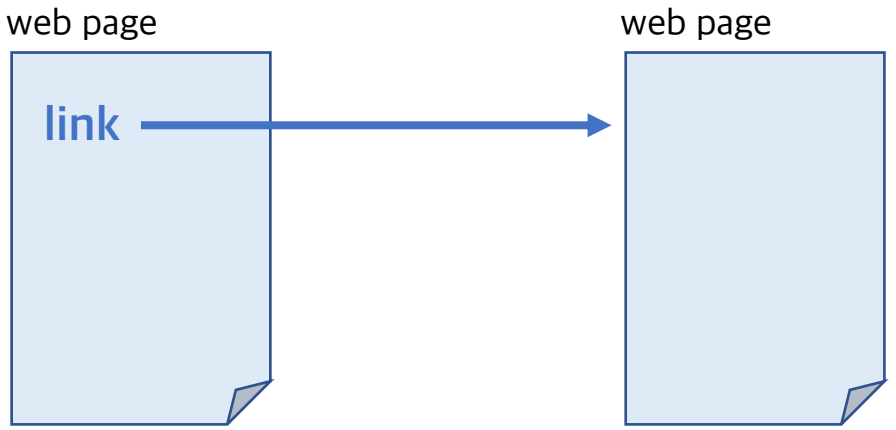
Spam 정보를 거르고 좋은 검색 결과를 얻는 방법이 필요 ➔ **Ranking**

Ranking의 방법: **Link Analysis**

- PageRank
- HITS

3. Link Analysis

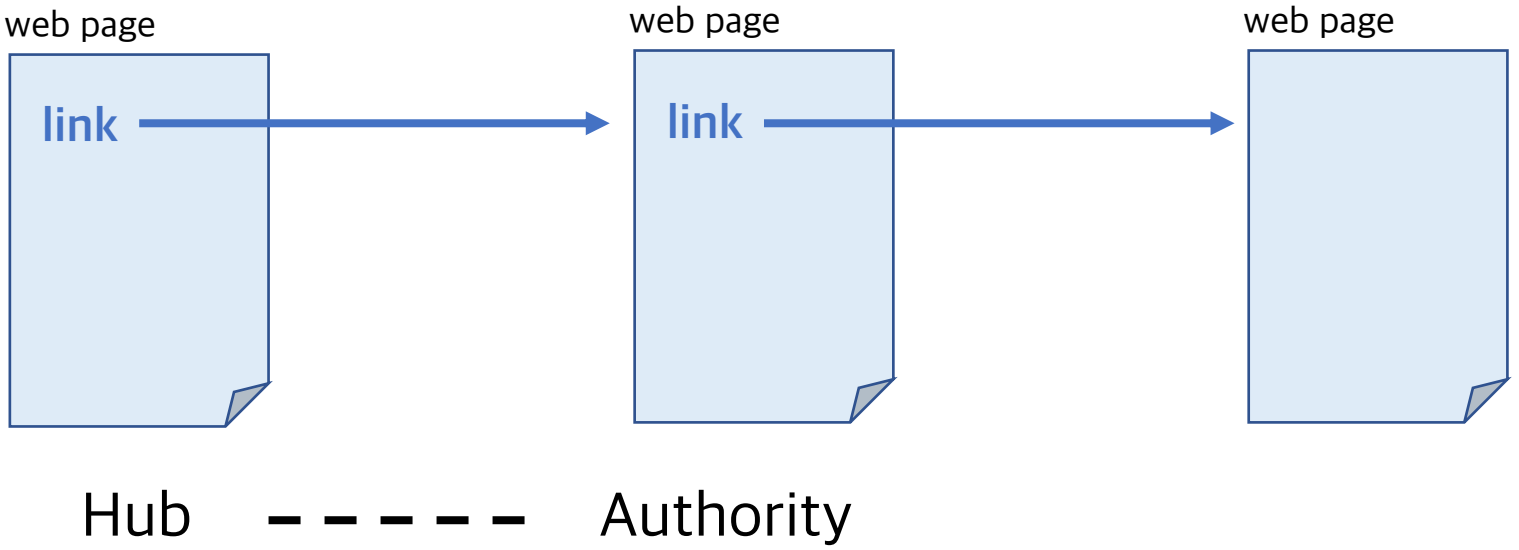
웹 페이지끼리 연결된 Link를 통해, 그 페이지가 얼마나 가치있는지 알 수 있다는 개념



Hub - - - - - Authority

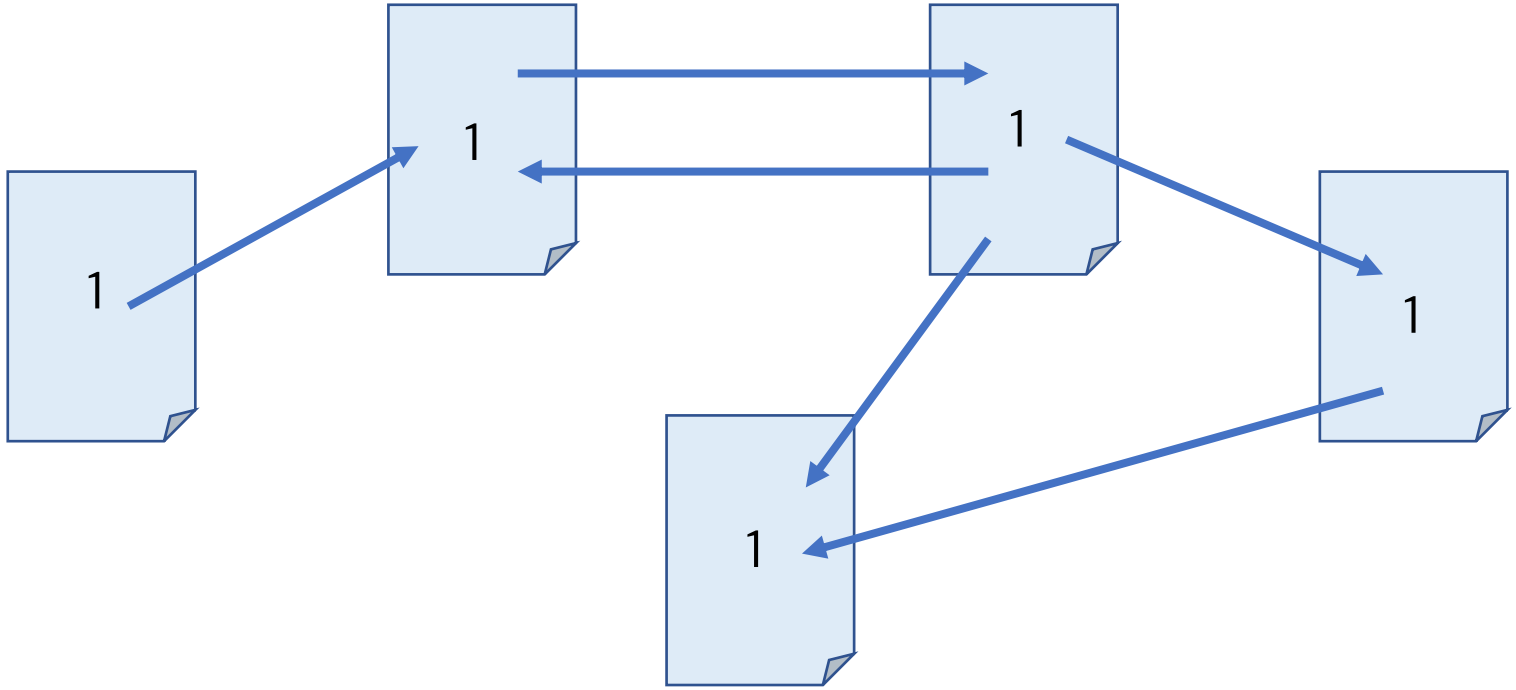
3. Link Analysis

웹 페이지끼리 연결된 Link를 통해, 그 페이지가 얼마나 가치있는지 알 수 있다는 개념



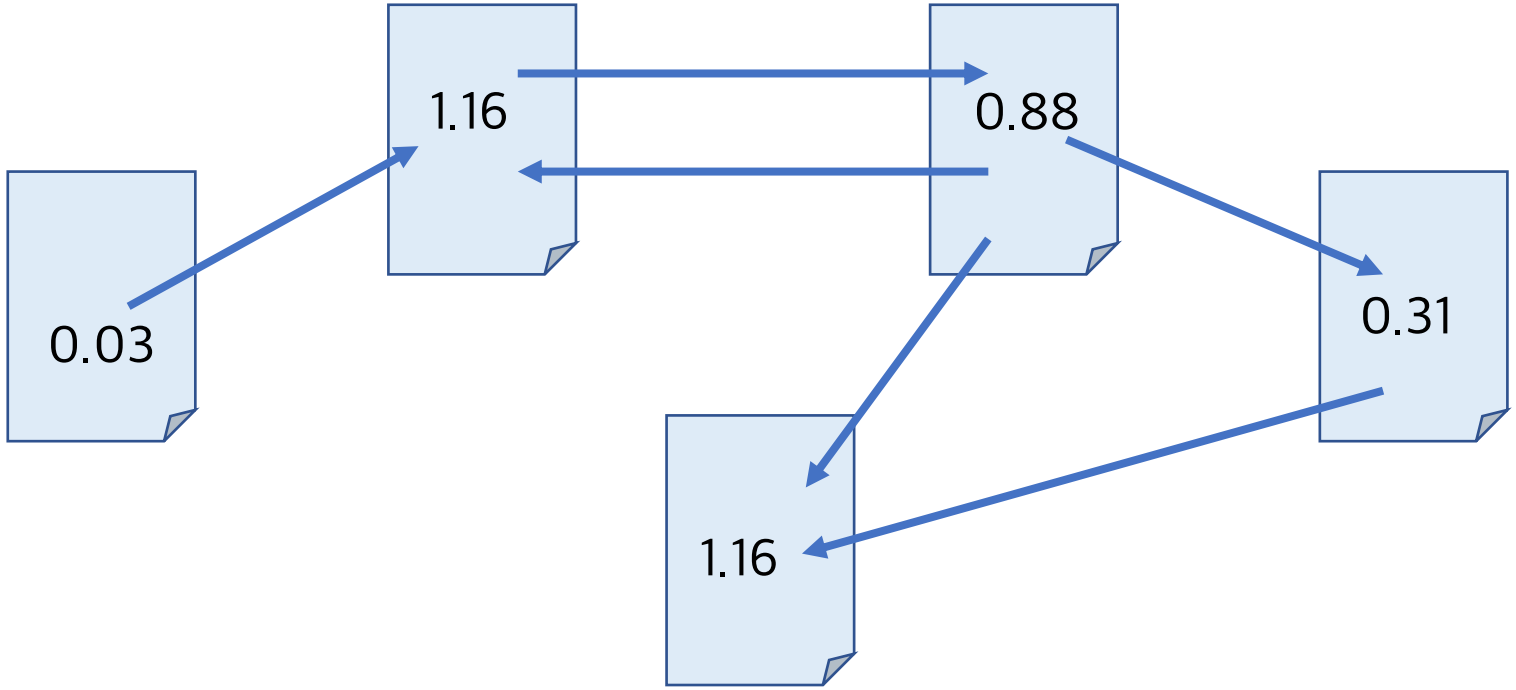
3. Link Analysis

각 페이지의 Rank 초기값을 지정한 후,



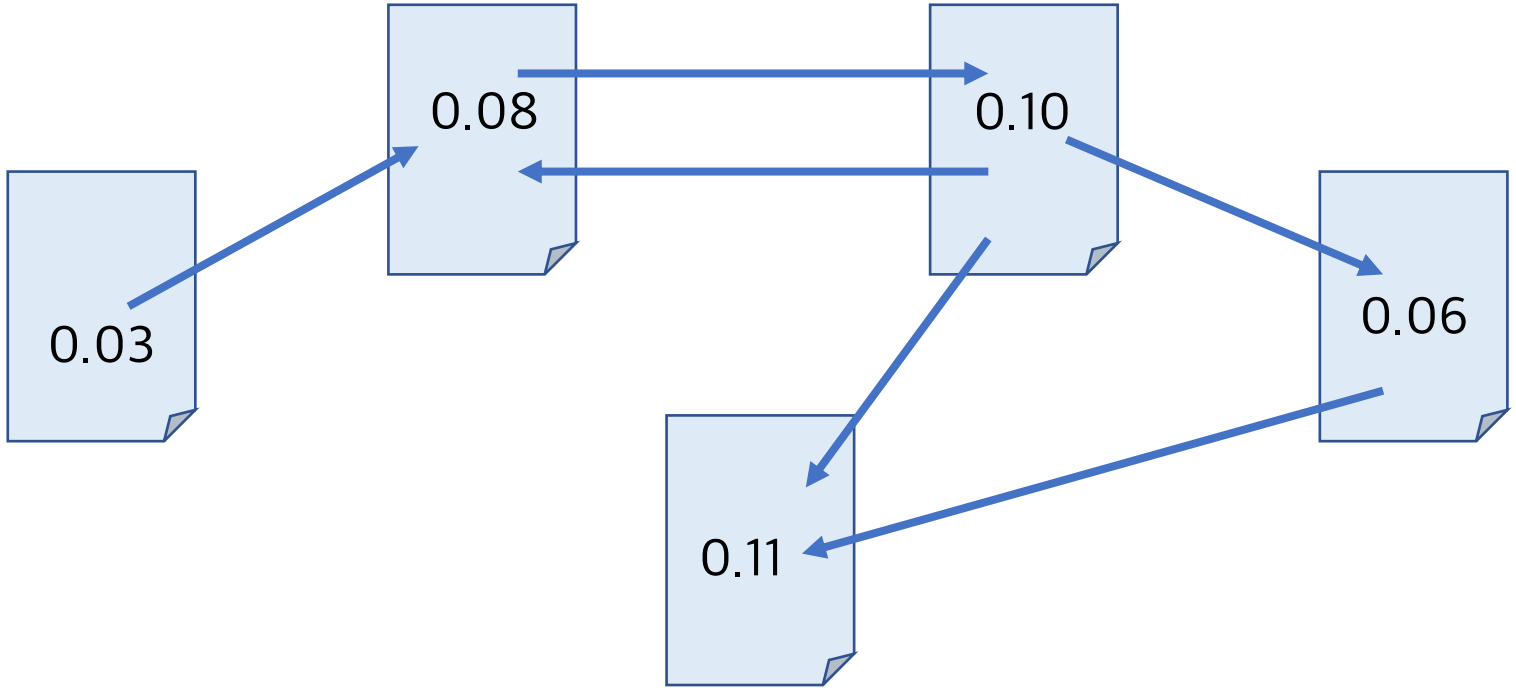
3. Link Analysis

각 페이지의 Rank 초기값을 지정한 후, Rank(i)와 Link에 의해 Rank(i+1) 계산



3. Link Analysis

각 페이지의 Rank 초기값을 지정한 후, Rank(i)와 Link에 의해 Rank(i+1) 계산



계산을 반복하면 수렴치에 도달함 ➔ 최종 Rank

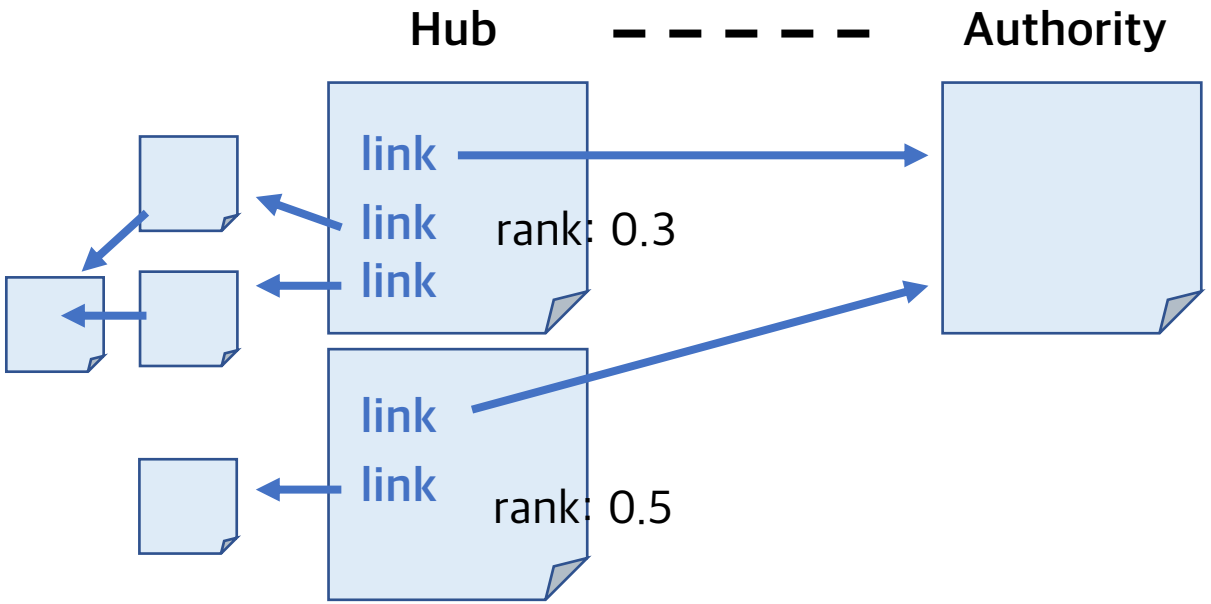
3. Link Analysis

- **PageRank**

$$Rank\ of\ Anthority = c \times \sum_{h}^{Hubs} \frac{Rank_h}{\#link\ of\ h} + \frac{1-c}{\#page}$$

Authority로의 link가 하나도 없을 때, bias역할

(c : [0,1]의 상수. PageRank 논문에서는 **0.85** 이용)



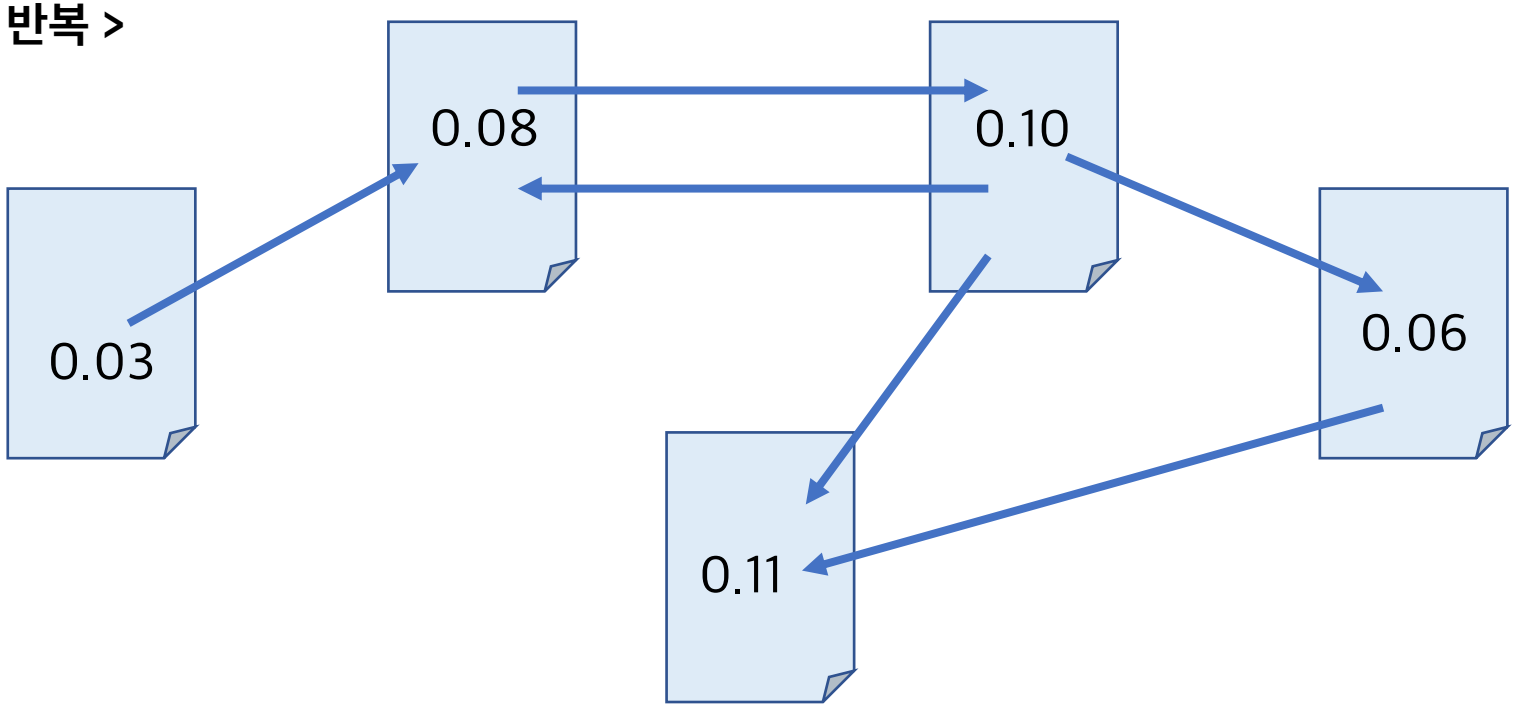
$$Rank = 0.85 \times \left(\frac{0.3}{3} + \frac{0.5}{2}\right) + \frac{1-0.85}{7}$$

$$= 0.32$$

3. Link Analysis

- **PageRank**

< 12회 반복 >



3. Link Analysis

- **HITS** (Hyperlink-Induced Topic Search)

Rank of Hub:

$$h_{i+1}(p) = \sum_{q:(q \to p)} a_i(q)$$

Rank of Authority:

$$a_{i+1}(p) = \sum_{q:(q \to p)} h_i(q)$$

Normalization:

$$\sum_p h_i(p)^2 = \sum_p a_i(p)^2 = 1$$

각 페이지의 h, a 값이 같아질 때까지

h, a 계산과 Normalization을 반복

4. Learning to Rank

- **Logistic Regression**
  문서가 사용자의 질의에 적절한 문서일 확률을 기계학습으로 계산
  (relevant or non-relevant 이므로 Logistic Regression 이용)

문서 d가 질의 q에 적절할 확률: $P(R = 1|q, d)$

$$\log \frac{P(R=1 \mid q,d)}{1-P(R=1 \mid q,d)} = \beta_0 + \sum_{i=1}^{n} \beta_i X_i. \qquad (10.5)$$

$$P(R=1 \mid d,q) = \frac{1}{1+\exp\left\{-\beta_0 - \sum_{i=1}^{n} \beta_i X_i\right\}}. \qquad (10.6)$$

d: 문서 q: 요청(질의) $R = 1$: 적절하다(relevant)
β: Weight X: Feature

4. Learning to Rank

- **Logistic Regression**

  Feature: TF-IDF, BM25, PageRank 등

	$X_1(q,d)$	$X_2(q,d)$	$X_3(q,d)$
$d_1(R=1)$	0.7	0.11	0.65
$d_2(R=0)$	0.3	0.05	0.4

Figure 10.9 Example of a combination of multiple features in ranking.

X1: query에 대한 BM25
X2: PageRank
X3: Anchor Text에 대한 BM25

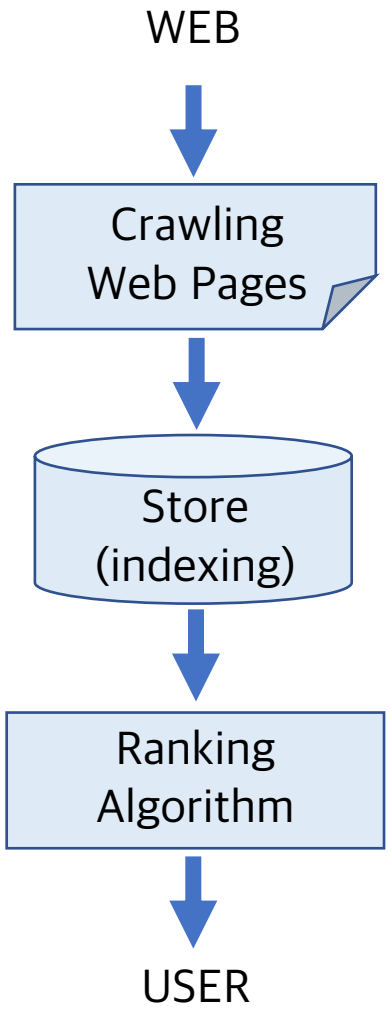
(Anchor Text: 링크를 설명하는 텍스트)
[구글로 이동] ← 이런 것

d1은 적절하고 d2는 적절하지 않을 확률:

$$p(\{q, d_1, R=1\}, \{q, d_2, R=0\}) = \frac{1}{1+\exp\{-\beta_0 - 0.7\beta_1 - 0.11\beta_2 - 0.65\beta_3\}} \times \left(1 - \frac{1}{1+\exp\{-\beta_0 - 0.3\beta_1 - 0.05\beta_2 - 0.4\beta_3\}}\right)$$

Conclusion

1. Web Crawling
2. Web Indexing
   - Google File System (store)
   - Map Reduce (compute)
3. Link Analysis
   - PageRank
   - HITS
4. Learning to Rank
   - Logistic Regression



Q&A

감사합니다.