

(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl.⁷
G06F 17/00
G06F 17/20

(11) 공개번호 10-2005-0027944
(43) 공개일자 2005년03월21일

(21) 출원번호 10-2004-0073658
(22) 출원일자 2004년09월15일

(30) 우선권주장 10/663,933 2003년09월16일 미국(US)
(71) 출원인 마이크로소프트 코포레이션
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원 마이크로소프트 웨이
(72) 발명자 나조르크,마크에이.
미국 94303 캘리포니아주 팔로 알토 툴립 레인 67
(74) 대리인 주성민
백만기
이중희

심사청구 : 없음

(54) 구조적으로 상호관련된 정보에 기초하여 문서를순위매김하기 위한 개선된 시스템 및 방법

요약

동족 등용(nepotistic)의 링크를 방지하는 방식으로 하이퍼링크 정보에 기초하여 웹 페이지를 순위매김하는 시스템 및 방법이 제공된다. 일 실시예에서, 웹 서치 서비스는 품질 질의 결과를 리턴하는 데 제공된다. 타겟 페이지(들)의 스코어를 올리기위해 인위적으로 생성되는 웹 페이지에 대한 PageRank와 같은 기존의 순위매김 알고리즘의 취약성을 다룬다. 직관적으로, 랜덤 점프를 통해 다수의 페이지를 갖는 웹 서버 상의 특정 페이지에 도달할 가능성은, 소수의 페이지를 갖는 웹 서버 상의 특정 페이지에 도달할 가능성보다 적으며, 이것은 다른 페이지로 링크하거나 다른 페이지를 보증함(endorsing)으로써 또다른 페이지에 대한 그러한 페이지의 영향이 감소됨을 의미한다는 것이 인지될 것이다. 따라서, 다양한 비제한적인 실시예에서, 각각의 웹 페이지가 아닌 각각의 웹 서버는 보장된 최소 스코어를 지정받는다. 서버에 지정된 이 최소 스코어는 그 웹 서버 상의 모든 페이지 사이에서 나뉘어 질 수 있다.

대표도

도 3e

색인어

웹, 문서, 서버, 링크, 순위매김, 스코어, 호스트, 인터넷 프로토콜

명세서

도면의 간단한 설명

도 1a 및 1b는 각각 정적 링크-기반 및 PageRank 알고리즘에 따른 순서매김 웹 페이지에 대한 종래 기술을 도시.
도 2a는 본 발명이 구현될 수 있는 다양한 컴퓨팅 장치를 갖는 예시적인 네트워크 환경을 나타내는 블록도.
도 2b는 본 발명이 구현될 수 있는 예시적인 비제한적인 컴퓨팅 장치를 나타내는 블록도.
도 3a 내지 3g는 본 발명에 따른 링크 스팸의 예방을 지지하는 직관적 행위를 도시.

도 4a 내지 4c는 서치 엔진 어플리케이션에서의 본 발명의 다양한 실시예의 예시적인 구현을 도시.

<도면의 주요 부분에 대한 부호의 설명>

110a, 110b, 110e - 컴퓨팅 장치

110c, 110d - 객체

10a, 10b - 서버 객체

20 - 데이터베이스

14 - 통신 네트워크/버스

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

<저작권 통지 및 허가>

본 특허 문서의 개시 부분은 저작권 보호에 속하는 특정 자료일 수 있다. 특허청 특허 파일 또는 기록에 나타나는 바와 같이, 저작권 소유자는 임의의 자에 의한 특허 문서 또는 특허 개시의 복제 재생산에 반대하지 않지만, 그렇지 않은 경우 모든 저작권을 소유한다. 다음의 통지는 본 명세에 적용된다: Copyright © 2003, Microsoft Corp.

본 발명은 구조적으로 상호관련된 정보에 기초하여 문서를 순위매김하는 것, 더 상세하게는, 동족 등용의 (nepotistic) 또는 스스로 서비스하는 링크를 방지하는 방식으로 하이퍼링크 정보에 기초하여 웹 페이지를 순위매김하는 것에 관한 것이다.

웹 서치 서비스(들)는 예를 들어, 사용자 또는 어플리케이션으로부터 질의를 수용하고, 결과 목록 예를 들어, 질의를 만족시키는 문서 또는 문서로의 링크를 리턴한다. 여기에 사용된 용어 "문서"는 검색될 수 있는 임의의 콘텐츠를 의미한다는 것을 유념해야 하며, 워드 프로세싱 문서 또는 웹 페이지와 같은 파일로 제한되는 것으로 해석해서는 안된다. 만족스러운 경험을 제공하기 위하여, 사용자와 가장 관련된 문서가 먼저 나타나야한다는 것을 고려하면, 이 결과 목록은 순서화되어야 한다. 문서를 순위매김하는 다수의 알고리즘이 현재 존재하고, 대부분의 웹 서치 엔진은 그러한 알고리즘 일부를 채용하여 상이한 순위매김 알고리즘에 의해 지정된 순위들의 조합에 기초하여 질의의 결과를 순위매김한다.

다수의 기존 순위매김 알고리즘은 그들이 질의-의존적(동적이라고도 불림)인지 질의-독립적(정적이라고도 불림)인지에 따라 분류될 수 있다. 질의-의존적인 순위매김 알고리즘이 질의 내의 용어를 사용하는 반면, 질의-독립적인 순위매김 알고리즘은 그렇지 않다. 즉, 질의-독립적인 순위매김 알고리즘은 웹 상의 각 문서에게 품질 스코어를 지정한다. 따라서, 질의-독립적인 순위매김 알고리즘은 유의하게도 미리 수행될 수 있고, 질의를 받을 때마다 재실행할 필요가 없다.

순위매김 알고리즘은 또한 콘텐츠-기반 순위 매김 알고리즘, 사용법-기반 순위 매김 알고리즘 및 링크-기반 순위매김 알고리즘으로 크게 분류될 수 있다. 콘텐츠-기반 순위매김 알고리즘은 문서 내의 단어를 사용하여 문서를 순위매김한다 (예를 들어, 질의-의존적인 콘텐츠-기반 순위매김 알고리즘은 문서 내의 더 초기에, 또는 크거나 강조된 폰트로 질의 용어를 포함하는 문서에게 더 높은 스코어를 줄 수 있다). 사용법-기반 순위매김 알고리즘은 얼마나 자주 관찰되는 지의 추정에 기초하여 웹 페이지를 순위매김한다. 그러한 추정은 웹 프록시 로그를 검사함으로써, 또는 서치 엔진의 결과 페이지 상의 사용자 클릭(click-through)을 모니터링함으로써 생성될 수 있다. 마지막으로, 링크-기반 순위매김 알고리즘은 웹 페이지들 사이의 하이퍼링크를 사용하여 웹 페이지를 순위매김한다.

예를 들어, 매우 단순한(naive) 정적 링크-기반 순위매김 알고리즘은 한 페이지를 가리키는 다른 페이지들로부터의 링크는 그 페이지를 보증(endorsing)한다는 아이디어로, 각각의 웹 페이지에게 그 페이지를 가리키는 링크(블랙링크)의 수에 비례하는 스코어를 지정할 수 있다. 예를 들어, 도 1a에서 웹 페이지 내의 검은 사각형에 의해 나타나는 바와 같이, 웹 페이지 A, B, C 및 D 각각은 다른 웹 페이지로의 링크(아웃링크) 3개를 포함한다. 이 예에서, 정적 링크-기반 순위매김 알고리즘을 사용하면, 페이지 D는 블랙링크를 가지고 있지 않기 때문에 페이지 C보다 낮은 스코어를 수신하고, 반면에, 페이지 C는 페이지 B로부터의 블랙링크 L2 한개, 및 페이지 A로부터의 블랙링크 L1 한개를 가지고 있다. 페이지 A, B, C 및 D를 다운로드 했을 때, 페이지는 관독될 수 있지만, 블랙링크 LU와 같이 알고리즘에 하나의 요인으로 포함될 수 없는 아직 알려지지 않은 임의의 위치로부터의 아직 알려지지 않은 임의의 블랙링크가 있을 수 있기 때문에, 각각 얼마나 많은 아웃링크를 가지고 있는 지, 및 어디로 링크하고 있는 지가 결정적이다. 이 단순한 접근의 주요 결점은 각각의 "보증"이 동일하게 취급되어, 사리를 위해 이용하기 쉬운 시스템으로 만든다는 것이다.

PageRank는 훨씬 가장 잘 알려진 질의-독립적인 링크-기반 순위매김 알고리즘이고, 그 원리가 여기에서 진술된다. PageRank는 재귀 계층을 시스템에 추가함으로써 도 1a의 단순한 정적 링크-기반 시스템의 원리를 토대로 형성된다. 도 1b에서 도시되는 바와 같이, 4개의 웹 페이지와 PageRank의 직감이 도시된다. PageRank로, 보증된 페이지에 스코어를 지정할 때, 보증하는 페이지의 스코어가 고려된다. 따라서, 웹 페이지 E(스코어 100)로부터의 보증

가중치는 웹 페이지 G에 주어진 스코어에 웹 페이지 F(스코어 9)로부터의 보증보다 더 많이 영향을 준다. 직관적으로, 그 보증들 사이에서 나누어지는 보증 페이지의 스코어에 대하여 생각할 수 있다.

수학적으로, PageRank 알고리즘의 직관적 행위는 다음과 같이 설명될 수 있다. 알려진 웹 페이지 집합 및 이들 사이의 링크는 정점 집합 V(각각의 정점은 웹 페이지와 대응함) 및 에지 집합 E(각각의 에지(u,v)는 페이지 u에서 페이지 v로의 하이퍼링크와 대응함)를 갖는 그래프를 유도한다고 가정하자. |V|는 집합 V의 크기를 나타내고, O(u)는 정점 u의 아웃-디그리(out-degree)(즉, 웹 페이지 u에 포함된 하이퍼링크의 수)를 나타내고, p는 0과 1 사이의

$$R(v) = \frac{p}{|V|} + (1-p) \sum_{(u,v) \in E} \frac{R(u)}{O(u)}$$

수(말하자면 0.15)라고 하자. 웹 페이지 v의 PageRank R(v)는 $\frac{p}{|V|} + (1-p) \sum_{(u,v) \in E} \frac{R(u)}{O(u)}$ 으로 정의된다.

PageRank 공식은 종종 다음과 같이 설명된다. 웹 상에서 랜덤하게 행보하는 웹 서퍼(surfer)를 상상해보자. 행보에 따른 매 단계에서, 서퍼는 다음의 알고리즘을 이용하여 하나의 웹 페이지에서 다른 웹 페이지로 이동한다: 임의의 가능성 p를 가지고, 서퍼는 균일하게 랜덤으로 웹 페이지를 선택하고 그곳으로 점프한다. 그렇지 않으면, 서퍼는 현재 페이지에서 밖으로 나가는 하이퍼링크들 중 하나를 균일하게 랜덤으로 선택하여 그것을 따른다. 이 메타포어(metaphor) 때문에, 숫자 p는 때때로 "점프 가능성(서퍼가 완전히 랜덤 페이지로 점프할 가능성)"이라고 불린다. 웹 서퍼가 가능성 p를 가지고 점프하고 |V| 웹 페이지가 있다면, 특정 페이지로 점프할 가능성은 p/|V|이다. 임의의 페이지는 점프함으로써 도달되기 때문에, 모든 페이지는 적어도 p/|V|만큼의 스코어를 보장받는다.

PageRank 스코어는 질의 결과를 순위매김하는 데 사용될 수 있다. 모든 다른 요인이 동일한 상태에서, PageRank를 채용하는 서치 엔진은 높은 PageRank 스코어를 갖는 페이지를 낮은 스코어를 갖는 페이지보다 더 상위 순위 매김할 것이다. 서치 엔진의 대부분의 사용자는 처음 소수의 결과만을 검사하기 때문에, 상업적인 웹 사이트의 운영자는 자신의 사이트로의 링크가 결과 목록에서 일찍 나타나는 것 즉, 자신의 웹 페이지가 높은 PageRank 스코어를 수신하는 것에 매우 관심을 갖는다. 다시 말하면, 상업적인 웹 사이트 운영자는 자신의 웹 사이트 상의 페이지들의 PageRank 스코어를 인위적으로 증가시킬 의욕을 갖는다.

PageRank 공식을 분석함으로써, 웹 페이지 v의 PageRank 스코어를 증가시키는 하나의 방법은 많은 다른 페이지들을 그곳에 링크시키는 것이 분명해진다. 이것은 웹 페이지가 자신의 아웃링크를 통해 다른 웹 페이지를 보증할 수 있다는 아이디어가 PageRank의 핵심이기 때문이다. v에 링크한 모든 페이지가 낮은 PageRank 스코어를 갖는다면, 각각의 개별 페이지는 매우 약간만 기여할 것이다. 그러나, 모든 페이지는 최소 p/|V|의 PageRank 스코어를 갖기를 보장받고 있기 때문에, 많은 그러한 낮은 품질 페이지로부터의 링크는 여전히 총 상당한 양만큼 기여할 수 있다. 이것은 PageRank 알고리즘의 취약성을 드러낸다.

발명이 이루고자 하는 기술적 과제

실제로, 이 PageRank의 취약성은 유일한 목적이 자신의 메인 홈페이지를 "보증(endorsing)"하는 것인 매우 큰 페이지 집합을 포함하는 웹 사이트에 의해 이용당하고 있다. 일반적으로, 이 보증 페이지는 보증되어야 하는 페이지로의 링크, 및 다른 보증 페이지로의 다른 링크를 포함한다. 모든 보증 페이지는 비행 중에 자동적으로 생성된다. 따라서, 웹 크롤러(crawler)는 임의의 보증 페이지에 걸쳐 막혔을 때 (보증 페이지가 다른 보증 페이지에 링크하는 사실 때문에) 더 많은 보증하는 페이지를 계속해서 다운로드하고, 그것에 의해 많은 양의 보증 페이지를 축적한다. 이 많은 양의 페이지 즉, 단일 페이지를 보증하는 그들 모두는 보증받고 있는 페이지의 PageRank 스코어를 인위적으로 올린다. PageRank 스코어를 인위적으로 올리는 데 사용되는 기술은 소위 "링크 스팸(link spamming)" 또는 "링크 스팸(link spam)"이라고 알려져 있다.

개인화된 PageRank 스코어가 특정 시각(perspective)으로부터의 웹의 관람을 생성할 수 있다는 것도 알려져 있다. 예를 들어, 사용자의 북마크를 취하고 사용자의 북마크에 있는 그 페이지들의 PageRank 스코어를 올림으로써, 개인화된 PageRank 스코어링 시스템이 달성된다. 본질적으로, 웹 페이지를 북마크로 지정하는 사용자는 그 웹 페이지를 사용자가 스코어링 시스템에 기초가 되길 원하는 것으로서 절대적으로 보증하였다. 다수의 "링크 스팸" 페이지는 말할 것도 없이, 사용자가 "링크 스팸" 페이지를 북마크로서 선택하는 것은 드물지만, 각각의 링크 스팸 웹 페이지와 관련된 최소 스코어가 아직 존재하기 때문에 개인화된 PageRank에 대한 아이디어는 링크 스팸 문제를 절대적으로 다루지 않는다.

따라서, 기본적인 아이디어는 완전하지만, PageRank의 결과는 동족 등용의(nepotistic) 링크에 의해 도입된 간섭을 받는다. 즉, 페이지 패밀리는 보증자 또는 피보증자의 실제 장점을 고려하지 않고 자기-보증 및 증진을 목적으로 생성될 수 있다. PageRank 스코어와 관련하여 링크 스팸의 문제가 존재한다고 알려지면, 해결책은 본 분야를 벗어난다.

따라서, 개선된 질의-독립적인 링크-기반 순위매김 알고리즘, 더 상세하게는, 동족 등용의 링크 효과(들)를 현저하게 감소시키는 개선된 순위매김 시스템 및 방법이 요구된다. 또한, 타겟 웹 페이지 피보증자(들)와 관련된 PageRank 스코어를 인위적으로 올리려는 목적으로 자기-보증하는 웹 페이지 패밀리를 생성하는 링크 스팸의 의욕을 감소시키는 개선된 순위매김 시스템 및 방법이 요구된다.

발명의 구성 및 작용

본 분야의 상술된 결점을 고려하여, 본 발명은 문서의 구조적 상호관련성에 대한 정보에 기초하여 문서를 순서매김하는 시스템 및 방법을 제공한다. 본 발명의 시스템 및 방법은 동족 등용의 링크를 방지하는 방식으로 하이퍼링크 정보에 기초하여 웹 페이지를 순위매김하기 위해 사용될 수 있다. 다양한 실시예에서, 본 발명은 품질 질의 결과를 리턴하는 웹 서치 서버에서 구현된다. 본 발명은 타겟 페이지(들)의 스코어를 올리는 것만을 목적으로 인위적으로 생성되는 웹 페이지에 대한 PageRank와 같은 기존 순위매김 알고리즘의 취약성을 다룬다. 직관적으로, 본 발명은 랜덤 점프를 통하여 다수의 페이지를 가지고 있는 웹 서버 상의 특정 페이지에 도달할 가능성이 소수의 페이지를 가

지고 있는 웹 서버 상의 특정 페이지에 도달할 가능성보다 낮고, 이것은 다른 페이지로 링크하거나 다른 페이지를 보증(endorsing)함으로써 또다른 페이지에 대한 그러한 페이지의 영향이 감소됨을 의미한다는 것을 인식한다. 따라서, 다양한 비제한적인 실시예에서, 본 발명은 각각의 웹 페이지가 아닌 각각의 웹 서버에게 보장된 최소 스코어를 지정한다. 서버에 지정된 이 최소 스코어는 웹 서버 상의 모든 페이지 사이에서 나뉘어질 수 있다.

본 발명의 다른 장점 및 특징이 이하에서 설명된다.

본 발명에 따른 개선된 순위매김 알고리즘을 제공하기 위한 시스템 및 방법이 첨부 도면을 참조하여 설명된다.

개요

언급된 바와 같이, PageRank 알고리즘은 웹 상의 모든 페이지를 단일 숫자, PageRank로 압축하는 대담한 태스크를 수행한다. PageRank는 그 콘텐츠에 상관없이 단지 웹의 그래프 구조 내의 위치에 기초하여 모든 웹 페이지에 대해 글로벌 순위매김하는 것이다.

PageRank를 사용하면, 서치 결과는 순서화되어 더 중요하고 중심적인 웹 페이지가 선호되도록 한다. PageRank를 지지하는 직감적인 행위는 그것이 웹 페이지 자신에게는 외부에 있는 정보(일종의 동등 리뷰를 제공하는 자신의 블랙링크)를 사용하는 것이다. 또한, "중요한" 페이지로부터의 블랙링크는 재귀적인 정의에 의해 평균 링크로부터의 블랙링크보다 더 중요한 것으로 간주된다.

개인화된 PageRank 스코어는 예를 들어, 사용자의 북마크를 취하고 그 사용자의 북마크에 있는 페이지의 PageRank 스코어를 올림으로써 특정 시각으로부터의 웹 관람을 생성할 수 있다. 그러나, 개인화된 PageRank는 각각의 링크 스펙 웹 페이지와 관련된 최소 스코어가 여전히 있기 때문에 링크 스펙밍 문제를 절대적으로 다루지 않는다. 따라서, 링크 스펙머는 각각 자신의 최소 PageRank 스코어를 갖고 서로서로 및 타겟 보증자 웹 페이지를 보증함으로써 타겟 피보증자 웹 페이지의 스코어를 인위적으로 올리는 단일 웹 서버 상의 복수의 웹 페이지를 (요구된다면 자동적으로) 여전히 생성할 수 있다. 링크 스펙머가 일반적으로 하나 이상 생성하는 복수의 웹 페이지는 (A)동일한 기호 호스트 이름을 갖거나, (B)동일한 도메인과 관련되거나, (C)동일한 IP 주소와 관련될 것이다.

본 발명은 랜덤 점프가 소수의 페이지를 갖는 웹 서버 상의 특정 페이지에 도달할 가능성보다 다수의 페이지를 갖는 웹 서버 상의 특정 페이지에 도달할 가능성이 적다는 것을 인식함으로써 이 취약성을 다룬다. 이것은 다른 페이지로 링크하거나 다른 페이지를 보증함으로써 또다른 페이지에 대한 그러한 페이지의 영향이 감소됨을 의미한다. 따라서, 다양한 비제한적인 실시예에서, 본 발명은 각각의 웹 페이지가 아닌 각각의 웹 서버에게 보장된 최소 스코어를 지정한다. 그리고 이 최소 스코어는 그 웹 서버 상의 모든 페이지 사이에서 나뉘어질 수 있다.

예시적인 네트워크 및 분산 환경

본 기술분야에 통상의 지식을 가진 자는, 본 발명이 컴퓨터 네트워크의 일부로서 또는 분산 컴퓨팅 환경에서 배치될 수 있는 임의의 컴퓨터 또는 다른 클라이언트나 서버 장치와 관련되어 구현될 수 있음을 이해할 수 있다. 이 점에서, 본 발명은 임의의 수의 메모리 또는 저장 유닛, 및 본 발명에 따라 문서를 순위매김하는 프로세스와 관련되어 사용될 수 있는 임의의 수의 저장 유닛 또는 볼륨에 발생하는 임의의 수의 어플리케이션 및 프로세스를 구비한 임의의 컴퓨터 시스템 또는 환경에 속한다. 본 발명은 원격 또는 로컬 저장장치를 구비한 네트워크 환경 또는 분산 컴퓨팅 환경에서 배치된 서버 컴퓨터 및 클라이언트 컴퓨터를 갖는 환경에 적용할 수 있다. 본 발명은 또한, 프로그래밍 언어 기능, 번역, 및 원격 또는 로컬 서비스와 관련된 정보를 생성, 수신 및 송신하는 실행 능력을 갖는 자립형(standalone) 컴퓨팅 장치에도 적용될 수 있다. 웹 페이지를 다운로드 및 분석하는 것은 특히 네트워크 및 분산 컴퓨팅 환경에서 동작하는 그 컴퓨팅 장치와 관련되기 때문에, 본 발명에 따른 순위매김 알고리즘 및 기술은 그 환경에서 매우 효과적으로 적용될 수 있다.

분산 컴퓨팅은 컴퓨팅 장치 및 시스템 간의 교환에 의해 컴퓨터 자원 및 서비스의 공유를 제공한다. 이 자원 및 서비스는 파일에 대한 정보, 캐시 저장 및 디스크 저장의 교환을 포함한다. 분산 컴퓨팅은 네트워크 연결성을 이용하여, 클라이언트가 전체 기업에 이득이 되도록 그들의 집합적인 힘에 영향을 주는 것을 허용한다. 이 점에서, 다양한 장치는 본 발명의 순위매김 알고리즘 및 프로세스를 포함할 수 있는 어플리케이션, 객체 또는 자원을 구비할 수 있다.

도 2a는 예시적인 네트워크 또는 분산 컴퓨팅 환경의 개략도를 제공한다. 분산 컴퓨팅 환경은 컴퓨팅 객체(10a, 10b 등) 및 컴퓨팅 객체 또는 장치(110a, 110b, 110c 등)를 포함한다. 이 객체는 프로그램, 방법, 데이터 저장, 프로그램 가능한 논리 등을 포함할 수 있다. 객체는 PDA, 텔레비전, MP3 플레이어, 퍼스널 컴퓨터 등과 같은 동일하거나 상이한 장치의 일부를 포함할 수 있다. 각각의 객체는 통신 네트워크(14)로 다른 객체와 통신할 수 있다. 이 네트워크 자체는 도 2a의 시스템에 서비스를 제공하는 다른 컴퓨팅 객체 및 컴퓨팅 장치를 포함할 수 있고, 복수의 상호 연결된 네트워크를 나타낼 수 있다. 본 발명의 양태에 따라, 각각의 객체(10a, 10b 등 또는 110a, 110b, 110c, 등)는 본 발명에 따른 순위매김 프로세스의 사용을 요구하기 위하여 API를 이용할 수 있는 어플리케이션 또는 다른 객체, 소프트웨어, 펌웨어 및/또는 하드웨어를 포함할 수 있다.

또한, 110c와 같은 객체가 다른 컴퓨팅 장치(10a, 10b 등 또는 110a, 110b, 110c, 등)에 대해 호스트될 수 있다는 것도 이해할 수 있다. 따라서, 서술된 물리적인 환경이 컴퓨터로서 연결된 장치를 도시할 수 있지만, 그러한 설명은 단지 예시적인 것일 뿐이며 물리적인 환경은 대안적으로 PDA, 텔레비전, MP3 플레이어 등과 같은 다양한 디지털 장치, 인터페이스와 같은 소프트웨어 객체, COM 객체 등을 포함하는 것으로 서술 또는 기재될 수 있다.

분산 컴퓨팅 환경을 지원하는 다양한 시스템, 컴포넌트 및 네트워크 구성이 있다. 예를 들어, 컴퓨팅 시스템은 로컬 네트워크 또는 광역 분산 네트워크에 의해 유선 또는 무선 시스템으로 함께 연결될 수 있다. 현재, 다수의 네트워크는, 광역 분산 컴퓨팅에게 기반구조를 제공하고 다수의 상이한 네트워크를 포함하는 인터넷에 연결되어 있다. 임의

의 기반구조는 예를 들어 본 발명에 따른 상호관련된 링크를 갖는 문서를 순위매김하는 데 흔히 있는 예시적인 통신을 위해 사용될 수 있다.

홈 네트워크 환경에서는, 전력선, 데이터(무선 및 유선), 음성(예를 들어, 전화) 및 엔터테인먼트 매체와 같은 각각 독특한 프로토콜을 지원할 수 있는 적어도 4개의 상이한 네트워크 송신 매체가 있다. 등(light) 스위치 및 가전기와 같은 대부분의 홈 제어 장치는 연결성을 위해 전력선을 사용할 수 있다. 데이터 서비스는 광대역(예를 들어, DSL 또는 Cable 모뎀)으로서 집안에 들어갈 수 있고, 무선(예를 들어, HomeRF 또는 802.11B) 또는 유선(예를 들어, Home PNA, Cat 5, Ethernet, 심지어 전력선) 연결성을 사용하여 집안 내에서 액세스 가능하게 된다. 음성 트래픽은 유선(예를 들어, Cat 3) 또는 무선(예를 들어, 셀폰)으로서 집안에 들어갈 수 있고, Cat 3 배선을 사용하여 집안 내에서 분산될 수 있다. 엔터테인먼트 매체 또는 다른 그래픽 데이터는 위성 또는 케이블을 통하여 집안에 들어갈 수 있고, 일반적으로 동축 케이블을 사용하여 집안에서 분산될 수 있다. IEEE 1394 및 DVI도 매체 장치의 클러스터에 대한 디지털 상호연결이다. 이 모든 네트워크 환경 및 프로토콜 기준으로서 나타날 수 있는 다른 환경들은 인터넷으로 외부 세계에 연결될 수 있는 인트라넷과 같은 네트워크를 형성하기 위해 상호연결될 수 있다. 요약하면, 데이터의 저장 및 전송에 대하여 다수의 상이한 소스가 존재하기 때문에 컴퓨팅 장치는 본 발명에 따른 순위매김 기술을 이용하는 프로그램 객체에 흔히 있는 액세스되거나 이용되는 데이터와 같은 데이터 공유 방법을 요구할 것이다.

인터넷은 흔히 컴퓨터 네트워킹 분야에서 잘 알려진 TCP/IP 프로토콜 슈트(suit)를 이용하는 네트워크 및 게이트웨이의 집합을 의미한다. TCP/IP는 "Transmission Control Protocol/Internet Protocol"의 머리글자이다. 인터넷은 사용자가 네트워크(들)를 통해 상호작용하고 정보를 공유하는 것을 허용하는 네트워킹 프로토콜을 실행하는 컴퓨터에 의해 상호연결된 지리적으로 분산된 원격 컴퓨터 네트워크 시스템으로 설명될 수 있다. 그러한 넓게 퍼진 정보 공유 때문에, 인터넷과 같은 원격 네트워크는, 전체적으로 개발자가 특수화된 동작 또는 서비스를 본질적으로 제약 없이 수행하기 위하여 소프트웨어 어플리케이션을 설계할 수 있는 개방 시스템으로 발전했다.

따라서, 네트워크 기반구조는 클라이언트/서버, 피어-투-피어 또는 하이브리드 아키텍처와 같은 네트워크 토폴로지의 호스트를 가능하게 한다. "클라이언트"는 관련되지 않은 다른 클래스 또는 그룹의 서비스를 사용하는 클래스 또는 그룹의 구성원이다. 따라서, 컴퓨팅에서, 클라이언트는 다른 프로그램에 의해 제공되는 서비스를 요구하는 프로세스 즉, 대강 명령 또는 태스크의 집합이다. 클라이언트 프로세스는 다른 프로그램 또는 서비스 자체에 대한 임의의 작업 상세사항을 "알" 필요 없이 요구된 서비스를 이용한다. 클라이언트/서버 아키텍처 특히, 네트워크 시스템에서, 클라이언트는 보통 다른 컴퓨터 예를 들어, 서버에 의해 제공되는 공유된 네트워크 자원을 액세스하는 컴퓨터이다. 도 2a의 예에서, 임의의 컴퓨터는 환경에 따라 클라이언트, 서버 또는 양자 모두로 생각할 수 있지만, 컴퓨터(110a, 110b 등)는 클라이언트로, 컴퓨터(10a, 10b 등)는 서버로 생각할 수 있고, 여기서 서버(10a, 10b 등)는 클라이언트 컴퓨터(110a, 110b 등)에서 복제될 데이터를 유지한다. 이 컴퓨팅 장치들 중 임의의 장치는 데이터를 처리할 수도 있고, 본 발명의 순위매김 기술을 포함할 수 있는 서비스 또는 태스크를 요구할 수도 있다.

서버는 일반적으로 인터넷과 같은 원격 또는 로컬 네트워크를 통해 액세스 가능한 원격 컴퓨터 시스템이다. 통신 매체를 통해 서로 통신하면서 클라이언트 프로세스는 제1 컴퓨터 시스템에서 활성화될 수 있고 서버 프로세스는 제2 컴퓨터 시스템에서 활성화될 수 있어, 분산 기능을 제공하고 복수의 클라이언트가 서버의 정보 수집 능력을 이용하게 한다. 본 발명의 순위매김 기술에 따라 이용되는 임의의 소프트웨어 객체는 다수의 컴퓨팅 장치 또는 객체에 분산될 수 있다.

클라이언트(들) 및 서버(들)는 프로토콜 계층(들)이 제공하는 기능을 사용하여 서로 통신한다. 예를 들어, HTTP(HyperText Transfer Protocol)는 WWW(World Wide Web) 또는 "웹"과 결합되어 사용되는 흔한 프로토콜이다. 일반적으로, IP(Internet Protocol) 주소와 같은 컴퓨터 네트워크 주소 또는 URL(Universal Resource Locator)과 같은 다른 참조는 서버 또는 클라이언트 컴퓨터를 서로 식별하는 데 사용될 수 있다. 네트워크 주소는 URL 주소로 나타날 수 있다. 통신은 통신 매체를 통해서 제공될 수 있다. 예를 들어, 클라이언트(들) 및 서버(들)는 고성능 통신을 위해 TCP/IP 연결(들)을 통해 서로 연결될 수 있다.

따라서, 도 2a는 서버가 네트워크/버스를 통해 클라이언트와 통신하는 본 발명이 채용될 수 있는 예시적인 네트워크 또는 분산 환경을 설명한다. 보다 상세하게는, 다수의 서버(10a, 10b 등)는 본 발명에 따라 휴대 컴퓨터, 핸드헬드 컴퓨터, 웹 클라이언트, 네트워크 가전기기, 또는 VTR, TV, 오븐, 등(light), 히터 등의 다른 장치와 같은 다수의 클라이언트 또는 원격 컴퓨터 장치(110a, 110b, 110c, 110d, 110e 등)를 갖는 LAN, WAN, 인트라넷, 인터넷 등일 수 있는 통신 네트워크/버스(14)를 통해 상호연결되어 있다. 따라서, 본 발명은 구조적으로 상호관련된 링크를 갖는 문서들의 순서매김을 구현하는 것이 바람직하다는 것과 관련되어 임의의 컴퓨팅 장치에 적용될 수 있음이 예상된다.

통신 네트워크/버스(14)가 인터넷인 네트워크 환경에서, 예를 들어, 서버(10a, 10b 등)는 클라이언트(110a, 110b, 110c, 110d, 110e 등)와 HTTP와 같은 다수의 알려진 프로토콜 중 임의의 프로토콜을 통해 통신하는 웹 서버일 수 있다. 서버(10a, 10b 등)는 분산 컴퓨팅 환경의 특징 대로, 클라이언트(110a, 110b, 110c, 110d, 110e 등)로서도 서비스할 수 있다.

통신은 적합한 유선 또는 무선일 수 있다. 클라이언트 장치(110a, 110b, 110c, 110d, 110e 등)는 통신 네트워크/버스(14)를 통해 통신할 수도 있고 안할 수도 있고, 그와 관련된 독립적인 통신을 가질 수 있다. 예를 들면, TV 또는 VCR의 경우에, 그 제어에 네트워크화된 측면이 있을 수도 있고 없을 수도 있다. 각각의 클라이언트 컴퓨터(110a, 110b, 110c, 110d, 110e 등) 및 서버 컴퓨터(10a, 10b 등)는 다양한 어플리케이션 프로그램 모듈 또는 객체(135)를 갖출 수 있고, 파일 또는 데이터 스트림이 저장될 수 있거나 파일 또는 데이터 스트림의 일부(들)가 다운로드, 전송 또는 이주될 수 있는 다양한 유형의 저장 요소 또는 객체로의 연결 또는 액세스를 갖출 수 있다. 임의의 하나 이상의 컴퓨터(10a, 10b, 110a, 110b 등)는 본 발명에 따라 처리된 데이터를 저장하기 위한 데이터베이스 또는 메모리(20)와 같은 데이터베이스(20) 또는 다른 저장 요소의 유지 및 갱신을 담당할 수 있다. 따라서, 본 발명은 컴퓨터 네트워크/버스(14)를 액세스하고 그것과 상호작용할 수 있는 클라이언트 컴퓨터(110a, 110b 등), 및 클라이언트 컴퓨터(110a, 110b 등)와 기타 유사한 장치 및 데이터베이스(20)와 상호작용할 수 있는 서버 컴퓨터(10a, 10b 등)를 갖는 컴퓨터 네트워크 환경에서 사용될 수 있다.

예시적인 컴퓨팅 장치

도 2b 및 다음 설명은 본 발명이 구현될 수 있는 것과 관련된 적합한 컴퓨팅 환경의 간단한 일반적 설명을 제공한다. 그러나, 핸드헬드, 휴대용 및 다른 컴퓨팅 장치 및 모든 유형의 컴퓨팅 객체는 그것이 컴퓨팅 환경 내의 웹 페이지 또는 다른 구조적으로 상호관련된 문서와 인터페이스로 연결된 어느 곳에서나 본 발명과 관련된 사용이 예상됨을 이해해야 한다. 범용 목적 컴퓨터가 이하에 설명되고 있지만, 이것은 단지 일례일 뿐이고, 본 발명은 네트워크/버스 상호동작성 및 상호작용을 갖는 썬 클라이언트로 구현될 수 있다. 따라서, 본 발명은 매우 작거나 극소의 클라이언트 자원이 포함된 네트워크화된 호스트되는 서비스 환경 예를 들어, 클라이언트 장치가 단지 가전기에 놓인 객체와 같이 네트워크/버스로의 인터페이스로서 서비스하는 네트워크화된 환경에서 구현될 수 있다. 본질적으로, 데이터가 저장될 수도 있고 검색될 수도 있고 다른 컴퓨터로 송신될 수도 있는 곳은 어디나 본 발명에 따른 순위매김 기술의 동작을 위한 바람직하거나 적합한 환경이다.

요구되는 것은 아니지만, 본 발명은 장치 또는 객체를 위해 서비스하는 개발자가 사용하도록 오퍼레이팅 시스템을 통해 구현될 수 있고/있거나 본 발명의 순위매김 기술과 관련되어 동작하는 어플리케이션 소프트웨어 내에 포함될 수 있다. 소프트웨어는 클라이언트 워크스테이션, 서버 또는 다른 장치와 같은 하나 이상의 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터 실행가능 명령의 일반적인 문맥에서 설명될 수 있다. 일반적으로 프로그램 모듈은 특정 태스크를 수행하거나 특정 추상 데이터 타입을 구현하는 루틴, 프로그램, 객체, 컴포넌트, 데이터 구조 등을 포함한다. 일반적으로, 프로그램 모듈의 기능은 다양한 실시예에서 요구되는 대로 결합되거나 분산될 수 있다. 또한, 본 기술분야에 숙련된 기술자는 본 발명이 다른 컴퓨터 시스템 구성 및 프로토콜과 함께 실시될 수 있음을 이해할 것이다. 본 발명과 사용하기에 적합할 수 있는 기타 잘 알려진 컴퓨팅 시스템, 환경 및/또는 구성은 퍼스널 컴퓨터(PC), ATM(automated teller machine) 서버 컴퓨터, 핸드헬드 또는 랩탑 장치, 멀티 프로세서 시스템, 마이크로프로세서 기반의 시스템, 프로그램가능한 전자제품, 네트워크 PC, 가전기기, 등, 환경적인 제어 요소, 미니컴퓨터, 메인프레임 컴퓨터 등을 포함하지만 이것으로 제한되는 것은 아니다. 본 발명은 또한 통신 네트워크/버스 또는 기타 데이터 송신 매체를 통해 링크되는 원격 프로세싱 장치에 의해 태스크가 수행되는 분산 컴퓨팅 환경에서도 실시될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 메모리 저장 장치를 포함하는 로컬과 원격 컴퓨터 저장 매체 양자 내에 위치될 수 있고, 클라이언트 노드는 차례로 서버 노드로서 행동할 수 있다.

따라서, 도 2b는 상기에서 명백해 졌지만 본 발명이 구현될 수 있는 적합한 컴퓨팅 시스템 환경(100)의 예를 도시하고 있으며, 컴퓨팅 시스템 환경(100)은 단지 적합한 컴퓨팅 환경의 일례일 뿐이며, 본 발명의 사용 또는 기능의 범위에 어떤 제한도 제시하지 않는다. 컴퓨팅 환경(100)은 예시적인 오퍼레이팅 환경(100)에서 도시되는 컴포넌트들 중 임의의 하나 또는 그 조합과 관련하여 어떤 종속성 또는 요구사항을 갖지 않는 것으로 해석되어야 한다.

도 2b를 참조하면, 본 발명을 구현하는 예시적인 시스템은 컴퓨터(110) 형태의 범용 컴퓨팅 장치를 포함한다. 컴퓨터(110)의 컴포넌트는 프로세싱 유닛(120), 시스템 메모리(130), 및 시스템 메모리에서 프로세싱 유닛(120)으로의 연결을 포함하여 다양한 시스템 컴포넌트를 연결하는 시스템 버스(121)를 포함할 수 있지만, 그것으로 제한되는 것은 아니다. 시스템 버스(121)는 메모리 버스 또는 메모리 제어기, 주변장치 버스, 및 임의의 다양한 버스 아키텍처를 사용하는 로컬 버스를 포함하는 임의의 몇몇 유형의 버스 구조일 수 있다. 예를 들면, 그러한 아키텍처는 ISA(Industry Standard Architecture) 버스, MCA(Micro Channel Architecture), EISA(Enhanced ISA) 버스, VESA(Video Electronics Standards Association) 로컬 버스, PCI(Peripheral Component Interconnect) 버스 (메자닌 버스로도 알려짐)를 포함하지만 그것으로 제한되는 것은 아니다.

컴퓨터(110)는 통상적으로 다양한 컴퓨터 판독가능 매체를 포함한다. 컴퓨터 판독가능 매체는 컴퓨터(110)에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수 있으며, 휘발성 및 비휘발성 매체, 분리형(removable) 및 비분리형(non-removable) 매체를 둘다 포함한다. 예로서, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체 및 통신 매체를 포함할 수 있지만, 이에 한정되는 것은 아니다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령, 데이터 구조, 프로그램 모듈 또는 다른 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 둘다 포함한다. 컴퓨터 저장 매체는 RAM, ROM, EEPROM, 플래쉬 메모리 또는 기타 메모리 기술, CD-ROM, DVD(digital versatile disk) 또는 기타 광학 디스크 저장장치, 자기 카세트, 자기 테이프, 자기 디스크 저장장치 또는 기타 자기 저장장치, 또는 컴퓨터(110)에 의해 액세스될 수 있고 원하는 정보를 저장하는 데 사용될 수 있는 임의의 기타 매체를 포함할 수 있지만, 이에 한정되지 않는다. 통신 매체는 통상적으로 반송파 또는 기타 전송 매체나 니즘 등의 변조된 데이터 신호에 컴퓨터 판독가능 명령, 데이터 구조, 프로그램 모듈, 또는 다른 데이터를 구현하며, 임의의 정보 전달 매체를 포함한다. "변조된 데이터 신호"라는 용어는 신호 내에 정보를 인코딩하도록 설정되거나 변환된 특성을 하나 또는 그 이상을 갖는 신호를 의미한다. 예로서, 통신 매체는 유선 네트워크 또는 직접 유선 접속 등의 유선 매체와, 음향, RF, 적외선 및 기타 무선 매체 등의 무선 매체를 포함하지만, 이에 한정되지 않는다. 상술한 것들 중의 임의의 조합이 컴퓨터 판독가능 매체의 범위 내에 포함되어야 한다.

시스템 메모리(130)는 ROM(131) 및 RAM(132) 등의 휘발성 및/또는 비휘발성 메모리의 형태의 컴퓨터 저장 매체를 포함한다. 시동중과 같은 때에 컴퓨터(110) 내의 구성요소들간에 정보를 전송하는 것을 돕는 기본 루틴을 포함하는 기본 입출력 시스템(133; BIOS)은 일반적으로 ROM(131)에 저장된다. RAM(132)은 일반적으로 프로세싱 유닛(120)에 즉시 액세스될 수 있고 및/또는 프로세싱 유닛(120)에 의해 현재 작동되는 프로그램 모듈 및/또는 데이터를 포함한다. 예로서, (한정하고자 하는 것은 아님) 도 2b는 오퍼레이팅 시스템(134), 어플리케이션 프로그램(135), 기타 프로그램 모듈(136), 및 프로그램 데이터(137)를 도시한다.

컴퓨터(110)는 또한 다른 분리형/비분리형, 휘발성/비휘발성 컴퓨터 저장 매체를 포함할 수 있다. 단지 예로서, 도 2b에는 비분리형 비휘발성 자기 매체로부터 판독하거나 그 자기 매체에 기록하는 하드 디스크 드라이브(140), 분리형 비휘발성 자기 디스크(152)로부터 판독하거나 그 자기 디스크에 기록하는 자기 디스크 드라이브(151), 및 CD-ROM 또는 기타 광학 매체 등의 분리형 비휘발성 광학 디스크(156)로부터 판독하거나 그 광학 디스크에 기록하는 광학 디스크 드라이브(155)가 도시되어 있다. 예시적인 오퍼레이팅 환경에서 사용될 수 있는 다른 분리형/비분리형, 휘발성/비휘발성 컴퓨터 저장 매체는 자기 테이프 카세트, 플래쉬 메모리 카드, DVD(Digital versatile disk), 디지털 비디오 테이프, 고체 RAM, 고체 ROM 등을 포함하지만 이에 한정되지 않는다. 하드 디스크 드라이브(141)는 일반적으로 인터페이스(140)와 같은 비분리형 메모리 인터페이스를 통해 시스템 버스(121)에 접속되고, 자기 디스크

크 드라이브(151) 및 광학 디스크 드라이브(155)는 일반적으로 인터페이스(150)와 같은 분리형 메모리 인터페이스에 의해 시스템 버스(121)에 접속된다.

앞서 기술되고 도 2b에 도시된 드라이브 및 그 관련 컴퓨터 저장 매체는 컴퓨터(110)를 위한 컴퓨터 판독가능 명령, 데이터 구조, 프로그램 모듈 및 기타 데이터의 저장을 제공한다. 도 2b에서, 예를 들어, 하드 디스크 드라이브(141)는 오퍼레이팅 시스템(144), 어플리케이션 프로그램(145), 기타 프로그램 모듈(146), 및 프로그램 데이터(147)를 저장하는 것으로 도시된다. 이들 컴포넌트는 오퍼레이팅 시스템(134), 어플리케이션 프로그램(135), 기타 프로그램 모듈(136), 및 프로그램 데이터(137)와 동일할 수도 있고 다를 수도 있다. 오퍼레이팅 시스템(144), 어플리케이션 프로그램(145), 다른 프로그램 모듈(146), 및 프로그램 데이터(147)는 최소한 다른 복사본(different copies)임을 나타내기 위하여 다른 번호를 부여하였다. 사용자는 일반적으로 마우스, 트랙볼, 또는 터치 패드라 불리는 포인팅 장치(161) 및 키보드(162)와 같은 입력 장치를 통해 컴퓨터(110)에 명령 및 정보를 입력할 수 있다. (도시되지 않은) 기타 입력 장치는 마이크로폰, 조이스틱, 게임 패드, 위성 안테나, 스캐너 등을 포함할 수 있다. 이들 입력 장치 및 그외의 입력 장치는 시스템 버스에 연결된 사용자 입력 인터페이스(160)를 통해 종종 프로세싱 유닛(120)에 접속되지만, 병렬 포트, 게임 포트 또는 유니버설 시리얼 포트(USB)와 같은 기타 인터페이스 및 버스 구조에 의해 접속될 수 있다. 노스브리지(NorthBridge)와 같은 그래픽 인터페이스(182)도 시스템 버스(121)에 접속될 수 있다. 노스브리지는 CPU 또는 호스트 프로세싱 유닛(120)과 통신하는 칩셋이고, AGP(accelerated graphics port) 통신에 대한 책임 가정한다. 하나 이상의 GPU(graphic processing unit)(184)는 그래픽 인터페이스(182)와 통신할 수 있다. 이 점에서, GPU(184)는 일반적으로 레지스터 저장장치와 같은 온-칩(on-chip) 메모리 저장장치를 포함하고, GPU(184)는 비디오 메모리(186)와 통신하며, 이 때 본 발명의 어플리케이션 변수가 충돌할 수 있다. 그러나, GPU(184)는 단지 보조프로세서(coprocessor)의 일례일 뿐이므로 다양한 보조프로세싱 장치가 컴퓨터(110)에 포함될 수 있고, 픽셀 및 정점 셰이더(vertex shader)와 같은 다양한 프로시저 셰이더를 포함할 수 있다. 모니터(191) 또는 다른 유형의 디스플레이 장치는 또한, 차례로 비디오 메모리(186)와 통신할 수 있는 비디오 인터페이스(190) 등의 인터페이스를 통해 시스템 버스(121)에 접속된다. 모니터(191)외에도, 컴퓨터는 또한 출력 주변 인터페이스(195)를 통해 접속될 수 있는 스피커(197) 및 프린터(196) 등의 기타 주변 출력 장치를 포함할 수 있다.

컴퓨터(110)는 원격 컴퓨터(180)와 같은 하나 이상의 원격 컴퓨터로의 논리적 접속을 이용한 네트워크 환경에서 동작할 수 있다. 원격 컴퓨터(180)는 퍼스널 컴퓨터, 서버, 라우터, 네트워크 PC, 피어(peer) 장치, 또는 기타 공통 네트워크 노드일 수 있으며, 비록 도 2b에는 메모리 저장 장치(181)만이 도시되어 있지만, 컴퓨터(110)에 관하여 상술한 구성요소 중 다수 또는 모든 구성요소를 일반적으로 포함할 수 있다. 도 2b에 도시된 논리적 접속은 근거리 네트워크(LAN; 171) 및 원거리 네트워크(WAN; 173)를 포함하지만, 그 외의 네트워크/버스를 포함할 수도 있다. 이러한 네트워크 환경은 사무실, 기업 광역 컴퓨터 네트워크(enterprise-wide computer network), 인트라넷, 및 인터넷에서 일반적인 것이다.

LAN 네트워크 환경에서 사용되는 경우, 컴퓨터(110)는 네트워크 인터페이스 또는 어댑터(170)를 통해 LAN(171)에 접속된다. WAN 네트워크 환경에서 사용되는 경우, 컴퓨터(110)는 일반적으로 인터넷 등의 WAN(173)을 통해 통신을 구축하기 위한 모뎀(172) 또는 기타 수단을 포함한다. 내장형 또는 외장형일 수 있는 모뎀(172)은 사용자 입력 인터페이스(160) 또는 기타 적절한 메카니즘을 통해 시스템 버스(121)에 접속될 수 있다. 네트워크 환경에서, 컴퓨터(110)에 관하여 도시된 프로그램 모듈 또는 그 일부분은 원격 메모리 저장 장치에 저장될 수 있다. 예로서 (한정하고자 하는 것은 아님), 도 2b는 메모리 장치(181)에 상주하는 원격 어플리케이션 프로그램(185)을 도시한다. 도시된 네트워크 접속은 예시적인 것이며, 컴퓨터들간의 통신 링크를 구축하는 그 외의 수단이 사용될 수 있다.

예시적인 분산 컴퓨팅 프레임워크 또는 아키텍처

다양한 분산형 컴퓨팅 네트워크는 퍼스널 컴퓨팅 및 인터넷을 중심으로 개발되어 왔으며 개발 중에 있다. 개인 및 비즈니스 사용자를 막론하고, 컴퓨팅 활동을 점점 웹 브라우저 지향적 또는 네트워크 지향적으로 만드는 어플리케이션 및 컴퓨팅 장치에 대하여 이음새없이 상호 동작이 가능하고 웹 사용가능한 인터페이스(seamlessly interoperable and web-enabled interface)가 제공된다.

예를 들어, MICROSOFT[®]의 관리된 코드 플랫폼 즉, .NET은 서버, 웹-기반 데이터 저장과 같은 빌딩-블록 서비스, 및 다운로드 가능한 장치 소프트웨어를 포함한다. 일반적으로, .NET 플랫폼은 (1) 전 범위의 컴퓨팅 장치들이 작업을 함께 수행하도록 하며 사용자 정보가 그들 모두에 자동적으로 업데이트되고 동기화되도록 할 수 있는 능력, (2) HTML보다는 XML의 더 많이 사용함으로써 인해 가능한, 웹 페이지에 대한 증가된 상호작용 능력, (3) 예를 들어, Office.NET와 같은 소프트웨어 또는 이메일 등의 다양한 어플리케이션의 관리를 위하여, 중앙 시작점(central starting point)으로부터 사용자에게로의 제품 및 서비스의 커스터마이징된 액세스 및 전달을 특징으로 갖는 온라인 서비스, (4) 사용자와 장치간의 정보의 동기화 뿐만 아니라 정보 액세스의 효율성과 용이성을 증가시킬 집중된 데이터 저장, (5) 이메일, 팩스, 및 전화 등의 다양한 통신 매체를 통합할 수 있는 능력, (6) 개발자에 대하여, 재사용가능한 모듈을 생성하여 프로그래밍 예러의 수를 감소시키고 생산성을 증가시키는 능력, 및 (7) 많은 다른 크로스-플랫폼 및 언어 통합 특징을 제공한다.

예시된 실시예는 컴퓨팅 장치 상에 상주하는 소프트웨어와 관련하여 기재되지만, 본 발명의 하나 이상의 부분은, 본 방법이 다른 분산 컴퓨팅 프레임워크에서도 .NET 코드와 같은 관리된 코드에 의해 가능해지는 모든 언어 및 서비스에 포함될 수 있거나, 지원될 수 있거나, 이를 통해 액세스될 수 있도록, 오퍼레이팅 시스템, 어플리케이션 프로그램 인터페이스(API) 또는 "미들 맨(middle man)" 객체, 제어 객체, 하드웨어, 펌웨어, 중간 언어 명령어 또는 객체 등을 통해 구현될 수 있을 뿐 아니라, 기타 분산형 컴퓨팅 프레임워크에서 구현될 수도 있다.

웹 페이지를 순위매김하기 위한 시스템 및 방법

중래 기술에서 상술한 바와 같이, 본 발명은 타겟 페이지(들)의 스코어를 올리는 것만을 목적으로 인위적으로 생성되는 웹 페이지에 대한 PageRank와 같은 기존 순위매김 알고리즘의 취약성을 다룬다. 웹 서퍼가 특정 서버로 랜덤하게 접근할 것을 가정하면, 본 발명은 다수의 페이지를 갖는 웹 서버 상의 특정 페이지에 서퍼가 "착륙(land)"할 가능성이 소수의 페이지를 갖는 웹 서버 상의 특정 페이지에 서퍼가 "착륙"할 가능성보다 적음을 인식한다. 이 원리를

적용하여, 순위매김 알고리즘을 적용할 때, 본 발명은 그들이 동일한 서버로부터 생성된 경우 보증의 영향을 감소시킨다. 이 영향의 감소를 달성하기 위하여, 다양한 비제한적인 실시예에서, 본 발명은 각각의 웹 페이지가 아닌 각각의 웹 서버에게 보장된 최소 스코어를 지정한다. 서버에게 지정된 이 최소 스코어는 그 웹 서버 상의 모든 페이지 사이에서 나뉘어질 수 있다. 따라서, 웹 서버 상의 복수의 페이지를 통해 임의의 다수 동족 등용의(nepotistic) 링크를 생성하는 것은, 웹 서버의 소수 페이지 상의 상대적으로 적은 수의 동족 등용의 링크보다 더 나은 "보증 가치"를 달성하지 못한다. 일반적으로, 임의의 다수 동족 등용의 링크는 (A) 동일한 기호 호스트 이름을 갖거나, (B) 동일한 도메인과 관련되거나, (C) 동일한 IP 주소와 관련된 것이다. 일 실시예에서, 본 발명은 사용자에게 품질 질의 결과를 리턴하기 위해 웹 서치 서비스에서 구현된다.

무엇이 웹 서버를 구성하는지의 여러 가능한 정의가 존재함을 유념한다. 웹 서버는 기호 호스트 이름(예를 들어, www.google.com) 또는 공통 도메인에 의해 정의될 수도 있고, 하나의(또는 여러개의) IP 주소(예를 들어, 207.46.134.222)에 의해 정의될 수도 있다. 이들 중 하나의 웹 서버 주소의 정의를 이용한 구현이 링크 스팸 문제를 다루는 한편, 나중 2개의 정의는, 무한개의 웹 페이지를 서비스하는 웹 서버를 구성하는 것이 가능하듯이 거의 무한개의 호스트 이름(RFC 1035는 호스트 이름을 최대 255 문자 길이로 제한하고, 각각의 문자는 글자, 디지트 또는 하이픈임. 따라서, 실제로는 "거의 무한"인 37^{255} 개의 가능한 호스트 이름이 있음)을 해결하는 DNS 서버를 형성하는 것이 가능하기 때문에 본 발명의 목적에 더욱 적합하고 링크 스팸머에게 더 좋은 방해물로서 행동한다. 다음 설명은 2개의 대안적인 실시예에 대하여 차례로 각각의 정의를 사용한다.

본 발명에 따른 순위매김 메트릭(metric)의 제1 실시예에서는, 웹 서버가 기호 호스트 이름에 의해 정의된다. $h(u)$ 는 URL u 의 호스트 이름 컴포넌트를 표시하고 H 는 모든 호스트 집합 즉, $H=\{h(v):v \in V\}$ 이다. $V^H(h)$ 는 호스트 h 에 의해 서비스되는 URL 집합 즉, $V^H(h)=\{v:v \in V \wedge h(v)=h\}$ 이다. 이 2개의 정의는 본 발명의 제1 실시예에 따른 제1 새로운 순위매김 메트릭 R^H 의 정의에서 사용되며, 이것은 다음과 같다.

$$R^H(v) = \frac{p}{|H||V^H(h(v))|} + (1-p) \sum_{(u,v) \in E} \frac{R^H(u)}{O(u)}$$

본 발명에 따른 순위매김 메트릭의 제2 실시예에서, 웹 서버는 도메인 이름에 의해 정의된다. $d(u)$ 는 URL u 의 도메인 이름 컴포넌트를 표시하고 D 는 모든 도메인 집합 즉, $D=\{d(v):v \in V\}$ 이다. $V^D(d)$ 는 도메인 d 에서 웹 서버에 의해 서비스되는 URL 집합 즉, $V^D(d)=\{v:v \in V \wedge d(v)=d\}$ 이다. 이 두개의 정의는 본 발명의 제2 실시예에 따른 제2 새로운 순위매김 메트릭 R^D 의 정의에서 사용되며, 이것은 다음과 같다.

$$R^D(v) = \frac{p}{|D||V^D(d(v))|} + (1-p) \sum_{(u,v) \in E} \frac{R^D(u)}{O(u)}$$

본 발명의 제3 실시예에서는, 웹 서버가 IP 주소 집합에 의해 정의된다. $A(u)$ 는 URL u 가 서비스될 수 있는 IP 주소 집합(즉, $h(u)$ 가 해결하는 IP 주소 집합)을 표시한다. A 는 모든 IP 주소의 집합 즉, $A = \bigcup_{v \in V} A(v)$ 이다. $V^A(a)$ 는 IP 주소 a 에 의해 서비스되는 URL 집합 즉, $V^A(a)=\{v:v \in V \wedge a \in A(v)\}$ 이다. 이 두개의 정의는 본 발명의 제3 실시예에 따른 새로운 제3 순위매김 메트릭 R^A 의 정의에서 사용되며, 이것은 다음과 같다.

$$R^A(v) = \frac{p}{|A| \sum_{a \in A(v)} |V^A(a)|} + (1-p) \sum_{(u,v) \in E} \frac{R^A(u)}{O(u)}$$

본 발명은 이제 도 3a 내지 도 3g를 참조하여 설명된다. 도 3a는 제2 노드 N2로의 링크를 갖는 제1 노드 N1(예를 들어, 문서 또는 웹 페이지)을 도시한다. PageRank의 직감은 N1이 N2로 링크함으로써 그것의 존재를 보증 또는 "입증"하기 때문에 들어오는 링크가 없는 노드보다 N2가 더 좋은 페이지라는 것이다. PageRank의 직감은 도 3b를 도 3a에 비교함으로써 더 상세히 설명될 수 있다. 도 3a에서 N1은 단일 노드 N2에만 링크하는 반면, 도 3b에서는 노드 N1은 9개의 노드 N2 내지 N10을 링크하기 때문에, N1이 도 3a의 노드 N1과 관련된 다른 웹 페이지를 무차별적으로 보증하는 것으로 나타나므로, PageRank는 N1의 보증 가치를 내린다. 도 3a의 노드 N1은 보다 선택적으로 다른 노드를 보증하기 때문에, 대응하는 N2는, N1이 N2로 링크하는 결과로(모든 다른 요인은 동일함) 도 3a에서 보다 높은 스코어를 지정받는다. PageRank의 직감은 또한, 도 3c를 도 3a에 비교함으로써 더 상세히 설명될 수 있다. 이 비교에서, 도 3c의 보증하는 노드 N1에 지정된 스코어는 도 3a의 보증하는 노드 N1에 지정된 스코어의 10배이다. 따라서, 보증하는 노드의 품질(스코어)은 도 3c에서 더 높기 때문에, 도 3c의 노드 N2에 지정된 스코어는 도 3a의 노드 N2에 지정된 스코어에 비해 더 높다. 이 직감의 결합은 PageRank 알고리즘을 산출한다.

그러나, 종래기술에서 언급했듯이, PageRank에서, 모든 모드는 얼마나 많은 다른 노드를 보증하는지, 또는 자신의 PageRank 스코어가 얼마나 작은지에 상관없이 최소 스코어를 수신하기 때문에, 도 3d에 도시된 시나리오에 취약하다. 이 최소 스코어가 작은 반면, 작은 스코어의 증가가 매우 현저해질 수 있다. 따라서, 수동적으로 또는 자동적으로 다수의 보증하는 페이지를 생성함으로써, 노드 N2의 소유자, 각각 보증하는 노드 N2, 및 도 3d에 도시된 것과 같은 일부 형식 또는 배열에서의 서로를 예를 들어, 노드 N2의 스코어를 인위적으로 올릴 수 있다. 대체로, 이것을 행하기 위하여 비용 절감적인 방식으로, 링크 스팸머는 www.foo.com/page1, www.foo.com/page2, www.foo.com/page3 등과 같은 동일한 호스트 상에 추가적인 웹 페이지를 생성하는 것은 거의 비용이 들지 않기 때문에, 일반적으로 링크 스팸 노드(LS1 내지 LS9 등) 각각을 동일한 호스트(예를 들어, www.foo.com) 상에 놓을 것이다. 따라서, 영리한 링크 스팸머는 이 기술을 통해 인기없는 웹 페이지를 웹에 노출시키는 것을 증가시킬 수 있

다. 무한 수의 페이지를 동적으로 생성 및 서비스하는 일부 웹 서버가 있기 때문에, 웹 페이지의 수는 무한이다. 명백히, 또다른 웹 페이지를 생성하는 데 드는 증가 비용은 낮다.

응답으로, 본 발명은 그러한 링크 스팸 기술의 사용을 곤란하게 한다. 일 실시예에서, 링크 스팸 노드(LS1 내지 LS9) 각각이 모두 동일한 호스트 `www.foo.com` 상에 있다고 가정하면, 본 발명은 LS1 내지 LS9 각각에 최소 스코어를 지정하는 것 대신에, 각각의 호스트에 최소 스코어를 지정하고 그 최소 스코어를 그 호스트 상의 각각의 노드 사이에서 분배한다. 따라서, 1000개의 링크 스팸 노드 LS1 내지 LS1000가 호스트 `www.foo.com` 상에 있는, 하나의 링크 스팸 노드 LS1이 호스트 `www.foo.com` 상에 있는, 노드 N2의 보증에 대한 공헌은 동일하다(모든 다른 요인이 동일하다고 가정함). 따라서, 도 3e에 도시된 바와 같이, 본 발명은 호스트 `www.foo.com`에 최소 스코어를 지정하고, 그 최소 스코어를 페이지 LS1 내지 LS9 사이에서 나눈다.

그러나, 단호한 링크 스파머는 본 발명의 이 변형을 알지 못할 수 있다. 링크 스파머는 유사한 품질 스코어를 갖는 다른 웹 페이지보다 많이 득점하기 위하여 그 또는 그녀의 웹 페이지와 관련된 품질 스코어를 상대적으로 적은 양만큼 증가시키기로 결정할 수 있다. 예를 들어, 웹 페이지가 일반적인 서치 엔진 결과의 순위에서 순서매김의 5위에서 1위로 오르면, 품질 스코어 내의 상대적으로 작은 변경은 링크 스파머에게 현저한 이익일 수 있다. 인터넷 도메인 네이밍(naming) 시스템의 현재 형태에서, 사용자는 각각의 도메인 이름에 대해 연회비(약 \$25)를 지불하고, 그 도메인 내에서 임의의 수의 기호 호스트 이름을 생성할 수 있다. 따라서, 링크 스파머는 작은 수의 도메인을 얻을 수 있고, 그 도메인 내에서 임의의 가능한 호스트 이름을 해결하도록 DNS 서버를 형성할 수 있다. 그리고, 스파머는 이 도메인 내에서 다수의 상이한 호스트로부터 나오는 링크 스팸 페이지를 제공할 수 있고, 그에 의해 매우 많은 수의 페이지의 최소 스코어를 축적함으로써 페이지를 보증하는 능력을 재확장한다. 이 시나리오는 도 3f에 도시되어 있고, 여기서 복수의 웹 페이지는 단호한 링크 스파머에 의해 생성되고, 그 각각은 유일한 기호 호스트 이름 `www1.foo.com`, `www2.foo.com`, `www3.foo.com` 등으로부터 생긴다. 그러나, 이 각각은 또한, 공통의 도메인 `foo.com`으로부터 생긴다. 따라서, 본 발명의 제2 실시예에서, 최소 스코어는 각각의 웹 페이지 또는 각각의 개별 호스트 이름이 아닌 각각의 개별 도메인 이름에 지정된다.

판명된 바와 같이, 도메인 이름은 비용이 들지만, 매우 비싼것은 아니다. 무한 수의 웹 페이지 및 잠재적으로 많은 수(37²⁵⁵)의 도메인 이름이 있지만, 실제로는 약 1천 500만 개의 도메인 이름이 존재한다. 새로운 호스트 이름을 생성하는 것은 무료인 반면(도메인 내에서 임의의 가능한 호스트 이름을 해결하도록 DNS 서버를 형성함으로써), 새로운 도메인 이름을 생성하는 것은 약 \$25의 비용이 든다. 따라서, 도메인 이름이 약 \$25로 획득될 수 있기 때문에, 단호한 링크 스파머에게 있어서 도 3g에 도시된 시스템을 구현하는 것은 가능하다. 도 3g에서, 단호한 링크 스파머는 링크 스팸 노드 LS1 내지 LS9 등을 생성하였지만, 이번에는, 링크 스파머는 각각의 노드를 자신의 기호 호스트 이름 `www.foo.com`, `www.goo.com`, `www.hoo.com` 등을 갖는 자신의 도메인 상에 놓았다. 그러나, 가능성은 링크 스파머가 `www.foo.com`, `www.goo.com`, `www.hoo.com` 등의 각각을 동일한 IP 주소에 놓는다는 것이다. 약 40억 개의 개별 IP 주소가 있다. IP 주소가 상대적으로 획득하기에 덜 비싸지만, 링크 스파머는 개별 IP 주소의 공급이 무한이 아니기 때문에 링크 스팸 웹 페이지 당 개별 IP 주소를 지정하는 시스템을 구현하지 않으려 한다. 따라서, 본 발명의 제3 실시예에서, 최소 스코어는, 링크 스파머가 링크 스팸 노드를 갖는 100개의 개별 도메인을 서버 상에 놓든지, 또는 링크 스파머가 단일 링크 스팸 노드를 갖는 1개의 개별 도메인을 서버 상에 놓든지, 노드 N2의 보증에 관한 효과는 동일하기 때문에(다른 모든 요인은 고정됨), 단호한 링크 스파머로부터 보호하면서, 각각의 웹 페이지, 각각의 호스트 이름, 또는 각각의 도메인이 아닌 각각의 개별 IP 주소에 지정된다.

본 발명은 임의의 구조적으로 상호관련된 문서를 순서매김하는 데 적용될 수 있지만, 도 4a(시스템 다이어그램), 4b 및 4c(흐름도)는 서치 엔진 어플리케이션을 위하여 웹 페이지를 순위매김하는 본 발명의 예시적인 어플리케이션을 도시한다. 본 발명의 알고리즘은 질의 독립적이기 때문에, 본 발명에 따른 문서 순위매김 프로세스는 문서에 지정된 스코어에 기초하여 문서를 요구하는 서치 엔진과 같은 어플리케이션과 독립적으로 일어날 수 있다. 예를 들어, 도 4a에 도시된 바와 같이, 하나 이상의 크롤러(들)(410)는 인터넷 상의 웹 페이지와 같은 구조적으로 상호관련된 문서의 소스(400)를 크롤링하고(crawl), 리포지토리(repository) 내의 저장장소에 대하여 문서에 관한 관련 정보 또는 문서를 검색(420)한다. 문서에 관한 관련 정보는 다른 소스(들)(415)로부터도 올 수 있다.

크롤러는 서치 엔진 인덱스에 대한 엔트리를 생성하기 위하여 웹 사이트를 방문하여 그 페이지 및 다른 정보를 관독하는 프로그램이다. 크롤러는 서버에서 서버로의 하이퍼텍스트 링크를 따르고 서치 기준을 기초로 정보를 인덱싱함으로써, 새로운 문서 및 새로운 사이트를 위치시키는 데 사용된다.

웹 상의 주요 서치 엔진은 모두 "스파이더(spider)", "엔트(ant)", "로봇("봇") 또는 "지능형 에이전트"로도 알려진 그러한 프로그램을 갖는다. 크롤러는 일반적으로 새로운 또는 갱신된 것으로서 자신의 소유자에 의해 서브밋된 사이트를 방문하도록 프로그래밍된다. 전체 사이트 또는 특정 페이지는 선택적으로 방문 및 인덱싱될 수 있다. 크롤러는 모든 페이지가 관독될 때까지 그 사이트 상의 다른 페이지로의 링크를 따르면서, 한번에 하나의 사이트, 하나의 페이지를 통해 크롤링하기 때문에 명백히 그 이름을 얻은 것이다. 일반적으로, 크롤러는 동시에 다수의 웹 사이트를 크롤링한다. 일반적으로, 크롤러는 SRE(Standard of Robot Exclusion)에서 지정된 웹 크롤러에 대한 예외 규칙에 충실하다.

도 4b는 흐름도를 통해 문서를 모으는 프로세스를 도시한다. 450에서, 웹 페이지가 발견되고 웹 페이지 및/또는 웹 페이지에 관한 정보가 크롤러(들)를 통해 수집된다. 이것은 화살표 표시된 것과 같이, 반복될 수 있거나 진행중인 프로세스일 수 있다. 460에서, 문서의 컬렉션 구조가 알려진 때의 임의의 주어진 지점에서, 본 발명은 수집된 문서 각각에 스코어를 지정하도록 적용될 수 있어서, 웹 페이지(또는 그 링크)의 리포지토리는 R^A 매트릭, R^D 매트릭 또는 R^H 매트릭을 사용하여 관련된 품질 스코어를 각각 구비하여 존재한다. 객체(430)는 API(432)를 통하여 리포지토리 내의 문서(420) 각각에 스코어를 지정하는 것 이상의 지능을 수행할 수 있다. 이하에 보다 상세히 설명되는 바와 같이, R^A, R^D 및/또는 R^H 매트릭이 특정 어플리케이션 또는 사용자에게 관하여 문서에 지정된 스코어의 품질을 높이기 위해 다른 매트릭과 조합될 수 있다는 것도 또한 이해될 수 있다.

반복적, 연속적 또는 주기적으로 갱신될 수 있는 문서 및 스코어(420)의 초기의 리포지토리가 생성되면, 서치 엔진의 예시적인 어플리케이션이 적용될 수 있다. 예를 들어, 서치 엔진(또는 다른 어플리케이션) 객체(440)는 470에서 사용자로부터 입력 질의를 수신할 수 있다. 480에서, 그 질의에 기초하여, 질의 기준에 따른 질의 조건을 포함하는 웹 페이지는 최고 품질의 문서가 처음으로 또는 더 두드러지게 사용자에게 디스플레이되도록 그 관련된 스코어에 따라 검색 및 순위매김될 수 있다. 예시적인 실시예에서, 도 4a에 도시되는 바와 같이, 질의는 API(434)를 통해 객체(430)와 인터페이스 접속이 되어 있는 서치 엔진 서버 객체(440)에 의해 수신된다. 객체(430)는 질의에 기초하여 API(432)를 통해 리포지토리로부터의 관련 웹 페이지(또는 웹 페이지로의 링크)(420)를 검색하고 순서화한다. 객체(430)는 사용자에게 디스플레이하기 위하여 순서화된 결과 리스트를 어플리케이션 객체(440)에 리턴한다. 대안적으로, 객체(440)는 관련된 스코어에 기초하여 결과의 순서화를 수행할 수 있다.

명확해진 바와 같이, 본 발명의 메트릭은 조합될 수도 있고, 전체 메트릭에의 다른 개선점과 조합될 수도 있다. 예를 들어, "신뢰된" 문서의 루트 집합으로, 임의의 메트릭으로 획득된 스코어의 전체 품질이 개선될 수 있다. 그러한 개선은 닐슨(Nielsen) 비율을 고려하는 것을 포함한다. 예를 들어, 최고 비율의 닐슨 웹 페이지는 다른 스코어를 측정하는 웹 페이지의 신뢰된 기초로서 사용될 수 있다. 또한, 닐슨 비율은 가중치를 받는 스코어를 지정하기 위해 본 발명의 메트릭(들)과의 조합에 사용될 수 있다. 또는, 본 발명의 메트릭(들) 결과는 변칙 결과를 발견하기 위해 닐슨 비율과 비교될 수 있다.

본질적으로, 웹 페이지와 같은 문서의 신뢰가치성에 관한 정보의 임의의 외부 소스는 본 발명의 메트릭(들) 결과와의 조합에, 또는 그 결과를 검증하는 데 사용될 수 있다. 다른 예는 ISP로부터의 사용자 정보 콜렉션에 기초하여 정보를 사용하는 것을 포함한다. 예를 들어, ISP는 사용자가 특정 웹 페이지를 얼마나 자주 방문하는 지에 대하여 닐슨 정보와 유사한 직접적인 정보를 수집할 수 있고, 이에 따라 사용법 패턴에 기초하여 품질, 인기도 또는 신뢰가치성에 스코어를 지정할 수 있다. ISP 프록시 로그는 이 목적으로 검사될 수 있다. 다른 예는 서치 엔진 상의 사람들을 관찰하는 것을 포함한다. 단지 웹 페이지가 수신한다는 이유로 최고 스코어는 사용자가 그 웹 페이지를 선택하기가 가장 쉽다는 것을 의미하지 않는다. 따라서, 서치 엔진 상의 사용자 행동은 웹 페이지의 품질을 입증하도록 서비스한다. 또한, 인간 편집자에게 예를 들어, 10,000개의 양호한 웹 페이지를 인증하는 일을 지정할 수 있다. 진행 중인 책임 동안, 신뢰된 웹 페이지로 알려짐으로 인해, 본 발명의 메트릭을 통한 보증 파워가 올라가도록 이 10,000개의 스코어가 올라갈 수 있다. 품질 웹 페이지에 대한 정보의 또다른 소스는 사용자의 북마크와 같은 사용자의 선호로부터 얻을 수 있다. 간략하게 말하면, 본 발명의 메트릭(들)은 가장 좋은 사용자 경험이 제공되는 것을 보장하기 위하여 임의의 다른 알려진 품질 메트릭과 조합될 수 있다. 유익하게도, 본 발명의 메트릭(들)을 포함하는 임의의 조합은 링크 스파머의 노력을 좌절시킬 것이다.

어플리케이션 및 서비스가 본 발명의 순위매김 시스템 및 방법을 사용하는 것을 가능하게 하는 본 발명을 구현하는 다수의 방법 예를 들어, 적합한 API, 툴 키트, 드라이버 코드, 오퍼레이팅 시스템, 제어, 자립형 또는 다운로드 가능한 소프트웨어 객체 등이 있다. 본 발명은 본 발명에 따른 순위매김 기술의 어플리케이션에 대하여 본 발명의 사용자 API(또는 다른 소프트웨어 객체)의 관점으로만 뿐만 아니라 웹 페이지 또는 웹 페이지와 관련된 구조 정보로 수신하는 소프트웨어 또는 하드웨어로부터 생각한다. 따라서, 여기에 개시된 본 발명의 다양한 구현은 전체 하드웨어, 일부 하드웨어 및 일부 소프트웨어, 및 소프트웨어인 양태를 가질 수 있다.

전술한 바와 같이, 본 발명의 예시적인 실시예는 다양한 컴퓨팅 장치 및 네트워크 아키텍처와 관련되어 기재되어 있다. 그 기본적인 개념은 구조적으로 상호관련된 문서를 순위매김하는 데 요구되는 임의의 컴퓨팅 장치 또는 시스템에 적용될 수 있다. 예를 들어, 본 발명의 알고리즘(들) 및 하드웨어 구현은, 장치 상의 분리된 객체로서, 다른 객체의 일부로서, 재사용 가능한 제어로서, 서버로부터 다운로드 가능한 객체로서, 장치 또는 객체와 네트워크 간의 "미들 맨"으로서, 분산 객체로서, 하드웨어로서, 메모리에서, 상술된 이들의 임의의 조합 등으로서 제공되는 컴퓨팅 장치의 오퍼레이팅 시스템에 적용될 수 있다. 예시적인 프로그래밍 언어, 이름 및 예가 다양한 선택을 대표하는 것으로 여기서 선택되었지만, 이 언어, 이름 및 예가 제한되는 것으로 의도되지 않는다. 본 기술분야에 통상의 지식을 가진 자는 본 발명의 다양한 실시예에 의해 달성되는 동일하거나 유사하거나 동등한 기능을 달성하는 객체 코드 및 방법을 제공하는 다수의 방법을 있음을 이해할 것이다.

상술된 바와 같이, 여기에 기재된 다양한 기술은 하드웨어 또는 소프트웨어, 또는 그 양자의 적절한 조합과 관련되어 구현될 수 있다. 따라서, 본 발명의 방법 및 장치, 또는 특정 양태 또는 그 일부는 플로피 디스켓, CD-ROM, 하드 드라이브, 또는 임의의 다른 기계-판독가능 저장 매체와 같은 실체 매체에서 구현되는 프로그램 코드(즉, 명령어)의 형식을 취할 수 있고, 여기서 프로그램 코드가 로딩되어 컴퓨터 상의 프로그램 코드와 같은 기계에 의해 실행될 때, 그 기계는 본 발명을 실행하는 장치 가 된다. 프로그램 가능한 컴퓨터 상의 프로그램 코드 실행의 경우에, 컴퓨터 장치는 일반적으로 프로세서, 프로세서에 의해 판독 가능한 저장 매체(휘발성 및 비휘발성 메모리 및/또는 저장 요소를 포함함), 적어도 하나의 입력 장치, 및 적어도 하나의 출력 장치를 포함한다. 예를 들어, 데이터 프로세싱 API의 사용, 재사용 가능한 제어 등을 통해 본 발명의 순위매김 기술을 구현 또는 이용할 수 있는 하나 이상의 프로그램은 컴퓨터 시스템과 통신하기 위하여 고레벨 프로그래밍 언어 또는 객체 지향 프로그래밍 언어로 구현되는 것이 바람직하다. 그러나, 프로그램(들)은 요구된다면, 어셈블리 또는 기계 언어로 구현될 수 있다. 임의의 경우에, 언어는 컴파일 또는 번역된 언어일 수 있고 하드웨어 구현과 조합될 수 있다.

본 발명의 방법 및 장치는 또한, 전기 배선 또는 케이블링과 같은 일부 송신 매체 상에서 광섬유 또는 임의의 다른 송신 형태를 통해 송신되는 프로그램 코드 형식으로 구현된 통신을 통해 실행될 수 있고, 여기서 프로그램 코드가 수신 및 로드되어 EPROM, 게이트 어레이, PLD(programmable logic device), 클라이언트 컴퓨터 등과 같은 기계에 의해 실행될 때, 기계는 본 발명을 실행하는 장치가 된다. 범용 프로세서 상에서 구현될 때, 프로그램 코드는 본 발명의 기능을 유발하도록 동작하는 독특한 장치를 제공하는 프로세서와 조합된다. 따라서, 본 발명과 관련되어 사용된 임의의 저장 기술 역시 하드웨어와 소프트웨어의 조합일 수 있다.

본 발명은 다양한 도면의 바람직한 실시예와 관련하여 기재되었지만, 다른 유사한 실시예가 사용될 수도 있고, 기재된 실시예에서 그로부터 벗어나지 않고 본 발명의 동일한 기능을 수행하는 변경 및 추가가 이루어 질 수도 있음을 이해할 것이다. 예를 들어, 본 발명의 예시적인 네트워크 환경은 피어-투-피어 네트워크 환경과 같은 네트워크 환경의 문맥에서 설명되지만, 본 기술분야에 숙련된 기술자는 본 발명이 이것에 제한되지 않고, 본 명세서에 기재된 방법이 게이밍 콘솔, 핸드헬드 컴퓨터, 휴대용 컴퓨터 등의 임의의 컴퓨팅 장치 또는 환경에 적용될 수도 있고, 무선이

든 유선 이든 통신 네트워크를 통해 접속되고 그 네트워크를 통해 상호작용하는 임의의 수의 그러한 컴퓨팅 장치에 적용될 수도 있다는 것을 이해할 것이다. 또한, 핸드헬드 장치 오퍼레이팅 시스템 및 다른 어플리케이션 특정 오퍼레이팅 시스템을 포함한 다양한 컴퓨터 플랫폼은 특히, 계속해서 증가하는 무선 네트워크 장치의 수로 예상됨을 강조한다.

예시적인 실시예가 웹 서치 서비스의 문맥에서 본 발명을 이용하는 것으로 나타나지만, 본 발명은 웹 서치 서비스의 문맥으로만 제한되지 않고, 오히려 임의의 문서 또는 콘텐츠 집합에 일종의 품질 메트릭을 제공하는 것으로 구현될 수 있는데, 이것은 어느 측면에서는 서로를 의미한다. 예를 들어, 사용자는 일부 측면에서 (사람, 장소, 시간, 이벤트, 아티스트, 앨범, 타이틀, 배우 등에 의해) 구조적으로 상호관련있는 자신의 컴퓨터(또는 복수의 컴퓨팅 장치)에 저장된 그림, 영화, 노래 등의 집합을 가질 수 있고, 본 발명의 스코어링이 콘텐츠의 구조적인 상호 관련성에 기초하여 그 그림에 적용될 수 있다. 예를 들어, 제1 콘텐츠와 제2 콘텐츠 사이의 유사성 또는 다른 관련성은 제1 콘텐츠에서 제2 콘텐츠로의 링크로 간주될 수 있다. 또한, API(432)가 스코어링 프로세스와 관련되어 상술되었지만, 개별 API가 그 목적으로 구현될 수 있다는 것 즉, 질의에 기초하여 검색 및 순서화하는 데 API(432)를 사용하는 것은 스코어링 프로세스와 연계되지 않아도 됨을 유념한다. 또한, 여기에서 사용된 용어 "~에 비례하여"는 2개의 엔터티 사이의 임의의 수학적 관계를 말하며, 여기서 하나의 엔터티가 증가하면 기하, 선형, 지수, 대수 및 다른 관계를 포함한 모두 알려진 수학적 관계에 따라 다른 엔터티가 증가하지만, 이것으로 제한되는 것은 아니다. 용어 "반비례" 또는 "반비례하여"에도 동일하게 적용된다. 즉, 하나의 엔터티가 증가하면 다른 엔터티가 감소한다. 또한, 용어 서버는 다양한 문맥에서 다양한 것을 의미할 수 있기 때문에, 여기에서 다양하게 사용되는 용어 "웹 서버"는 적어도 (A) 동일한 기호 호스트 이름을 갖는 복수의 웹 페이지, (B) 동일한 도메인과 관련된 복수의 웹 페이지, 및 (C) 동일한 IP 주소와 관련된 복수의 웹 페이지 중 임의의 하나 이상을 포함하는 서버 컴퓨터(들) 및/또는 서버 객체(들)를 의미한다. 또한, 본 발명은 복수의 프로세싱 칩 또는 장치에서 또는 그것을 통해 구현될 수 있고, 저장장치는 복수의 장치를 통해 유사하게 효력을 나타낼 수 있다. 따라서, 본 발명은 임의의 단일 실시예로 제한되어서는 안되며, 오히려 첨부 청구범위에 따른 범위 및 범주 내에서 형성되어야 한다.

발명의 효과

본 발명은 문서의 구조적 상호관련성에 대한 정보에 기초하여 문서를 순서매김하는 시스템 및 방법을 제공한다. 본 발명의 시스템 및 방법은 동족 등용의(nepotistic) 링크를 방지하는 방식으로 하이퍼링크 정보에 기초하여 웹 페이지를 순위매김하기 위해 사용될 수 있다.

(57) 청구의 범위

청구항 1.

구조적으로 링크된 복수의 문서들 중 하나의 문서에 스코어를 지정 - 상기 문서는 (A) 동일한 기호 호스트 이름을 갖는 복수의 웹 페이지를 포함하는 서버, (B) 동일한 도메인과 관련된 복수의 웹 페이지를 포함하는 서버, 및 (C) 동일한 IP 주소와 관련된 복수의 웹 페이지를 포함하는 서버 중 적어도 하나에 의해 정의되는 웹 서버 상에 위치하고, 상기 문서는 어디에 위치되든 간에 상기 구조적으로 링크된 복수의 문서들 중 적어도 하나의 다른 문서로부터의 적어도 하나의 백링크(backlink)를 가짐 - 하는 방법에 있어서,

상기 웹 서버에 위치하는 문서의 수에 반비례하여 문서에 상기 스코어를 지정하는 단계를 포함하는 방법.

청구항 2.

제1항에 있어서,

상기 적어도 하나의 다른 문서의 수에 비례하여 상기 문서에 스코어를 지정하는 단계를 더 포함하는 방법.

청구항 3.

제1항에 있어서,

상기 적어도 하나의 다른 문서 중 적어도 하나에 지정된 적어도 하나의 스코어에 비례하여 스코어를 지정하는 단계를 더 포함하는 방법.

청구항 4.

제1항에 있어서,

(A) 상기 적어도 하나의 다른 문서의 수, 및 (B) 상기 적어도 하나의 다른 문서 중 적어도 하나에 지정된 적어도 하나의 스코어에 비례하여 상기 스코어를 지정하는 단계를 더 포함하는 방법.

청구항 5.

제2항에 있어서,

상기 적어도 하나의 다른 문서 중 적어도 하나의 아웃링크(outlink) 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 단계를 더 포함하는 방법.

청구항 6.

제1항에 있어서,

상기 지정 단계는, 상기 문서와 동일한 도메인 상에 위치하는 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 단계를 포함하는 방법.

청구항 7.

제1항에 있어서,

상기 지정 단계는, 상기 문서와 동일한 기호 호스트 이름을 갖는 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 단계를 포함하는 방법.

청구항 8.

제1항에 있어서,

상기 지정 단계는, 상기 문서와 동일한 인터넷 프로토콜(IP) 주소와 관련된 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 단계를 포함하는 방법.

청구항 9.

제1항에 있어서,

상기 제1 문서에 링크하는 적어도 하나의 다른 문서의 스코어를 합산하는 것에 기초하여 상기 문서에 상기 스코어를 지정하는 단계를 더 포함하는 방법.

청구항 10.

제1항에 있어서,

상기 구조적으로 링크된 복수의 문서들은 하이퍼링크를 갖는 웹 페이지이고, 상기 문서는 웹 페이지인 방법.

청구항 11.

제1항에 있어서,

상기 문서의 스코어를 웹 서치 서비스의 컴포넌트로 출력하는 단계를 더 포함하는 방법.

청구항 12.

제1항에 있어서,

평균 최소 스코어보다 더 높은 스코어를 선호되는 문서 집합에 지정하는 단계를 더 포함하는 방법.

청구항 13.

제12항에 있어서,

상기 선호되는 문서 집합은 닐슨 등급(Nielsen rating), 인간에 의해 지정된 비율 등급, ISP 프록시 로그로부터 추출된 웹 페이지 사용 패턴, 서치 엔진으로부터 추출된 웹 페이지 사용 패턴, 및 사용자 선호에 따라 지정된 문서 중 적어도 하나에 기초한 방법.

청구항 14.

제1항에 있어서,

제2 스코어링 기술에 기초하여 상기 문서의 스코어를 변경하는 단계를 더 포함하는 방법.

청구항 15.

제1항에 있어서,

변칙 결과를 발견하기 위하여 제2 스코어링 기술에 대해 상기 스코어를 비교하는 단계를 더 포함하는 방법.

청구항 16.

제1항의 방법을 수행하는 컴퓨터 실행가능 명령을 갖는 컴퓨터 실행가능 모듈을 포함하는 어플리케이션 프로그램 인터페이스.

청구항 17.

제1항의 방법을 수행하는 수단을 포함하는 컴퓨팅 장치.

청구항 18.

제1항의 방법을 수행하는 컴퓨터 실행가능 명령을 보유하는 변조된 데이터 신호.

청구항 19.

구조적으로 링크된 복수의 문서들 중 하나의 문서에 스코어를 지정하는 방법에 있어서,

상기 문서는 (A) 동일한 기호 호스트 이름을 갖는 복수의 웹 페이지를 포함하는 서버, (B) 동일한 도메인과 관련된 복수의 웹 페이지를 포함하는 서버, 및 (C) 동일한 IP 주소와 관련된 복수의 웹 페이지를 포함하는 서버 중 적어도 하나에 의해 정의되는 웹 서버 상에 위치하고, 상기 문서는 상기 구조적으로 링크된 복수의 문서들 중 적어도 하나의 소스 문서로부터의 적어도 하나의 백링크(backlink)를 가지고;

상기 문서의 스코어는 상기 적어도 하나의 소스 문서 중 적어도 하나와 관련된 적어도 하나의 스코어에 비례하여 계산되고;

상기 스코어는 상기 웹 서버 상에 위치한 상기 적어도 하나의 소스 문서의 수에 반비례하여 계산되는 방법.

청구항 20.

제19항에 있어서,

상기 스코어는, 상기 동일한 웹 서버 상에 위치한 상기 적어도 하나의 소스 문서의 수에 반비례하여 계산되는 방법.

청구항 21.

제20항에 있어서,

상기 스코어는, 동일한 기호 호스트 이름을 갖는 상기 적어도 하나의 소스 문서의 수에 반비례하여 계산되는 방법.

청구항 22.

제20항에 있어서,

상기 스코어는, 상기 동일한 도메인과 관련된 상기 적어도 하나의 소스 문서의 수에 반비례하여 계산되는 방법.

청구항 23.

제20항에 있어서,

상기 스코어는, 상기 동일한 인터넷 프로토콜(IP) 주소와 관련된 상기 적어도 하나의 소스 문서의 수에 반비례하여 계산되는 방법.

청구항 24.

제19항에 있어서,

상기 구조적으로 링크된 복수의 문서는 하이퍼링크를 갖는 웹 페이지이고, 상기 문서는 웹 페이지인 방법.

청구항 25.

제19항의 방법을 수행하는 컴퓨터 실행가능 명령을 갖는 컴퓨터 실행가능 모듈을 포함하는 어플리케이션 프로그래밍 인터페이스.

청구항 26.

제19항의 방법을 수행하는 수단을 포함하는 컴퓨팅 장치.

청구항 27.

제19항의 방법을 수행하는 컴퓨터 실행가능 명령을 보유하는 변조된 데이터 신호.

청구항 28.

질의 요구에 기초하여 웹 페이지 정보 및 관련된 스코어의 데이터베이스에 질의하는 질의 메카니즘 - 상기 데이터베이스로부터 검색된 결과는 각각의 결과와 관련된 스코어에 의해 순서화 되고, 상기 스코어는 상기 질의를 만족시키는 웹 페이지의 품질을 반영함 - 을 포함하는 서치 엔진과의 사용을 위한 어플리케이션 프로그래밍 인터페이스; 및

상기 웹 페이지 정보와 관련된 스코어를 생성하는 스코어링 객체 - 적어도 하나의 대응하는 소스 웹 페이지로의 적어도 하나의 백링크를 갖는 웹 페이지에 있어서, 상기 스코어링 객체는 상기 적어도 하나의 대응하는 소스 웹 페이지 중 적어도 하나와 관련된 적어도 하나의 스코어에 비례하여 상기 웹 페이지에 스코어를 지정하고, 상기 스코어는 동일한 웹 서버 상에 위치한 상기 적어도 하나의 대응하는 소스 웹 페이지의 수에 반비례하여 계산됨 -

를 포함하는 서버 객체.

청구항 29.

제28항에 있어서,

상기 스코어링 객체는 상기 질의 메카니즘과 독립적으로 동작하는 서버 객체.

청구항 30.

제28항에 있어서,

웹 서버가 공통의 기호 호스트 이름에 기초하여 정의되는 서버 객체.

청구항 31.

제28항에 있어서,

웹 서버가 공통의 도메인에 기초하여 정의되는 서버 객체.

청구항 32.

제28항에 있어서,

서버가 공통의 인터넷 프로토콜(IP) 주소에 기초하여 정의되는 서버 객체.

청구항 33.

구조적으로 링크된 복수의 문서들 중 하나의 문서에 스코어를 지정 - 상기 문서는 웹 서버 상에 위치하고 상기 구조적으로 링크된 복수의 문서들 중 적어도 하나의 다른 문서로부터 적어도 하나의 백링크를 가짐 - 하는 컴퓨터 실행가능 명령을 갖는 컴퓨터 실행가능 모듈을 포함하는 컴퓨터 판독가능 매체에 있어서,

상기 모듈은 상기 웹 서버에 위치한 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 수단을 포함하는 컴퓨터 판독가능 매체.

청구항 34.

제33항에 있어서,

상기 적어도 하나의 다른 문서의 수에 비례하여 상기 문서에 상기 스코어를 지정하는 수단을 더 포함하는 컴퓨터 판독가능 매체.

청구항 35.

제33항에 있어서,

상기 적어도 하나의 다른 문서 중 적어도 하나에 지정된 적어도 하나의 스코어에 비례하여 상기 스코어를 지정하는 수단을 더 포함하는 컴퓨터 판독가능 매체.

청구항 36.

제33항에 있어서,

(A) 상기 적어도 하나의 다른 문서의 수, 및 (B) 상기 적어도 하나의 다른 문서 중 적어도 하나에 지정된 적어도 하나의 스코어에 비례하여 상기 스코어를 지정하는 수단을 더 포함하는 컴퓨터 판독가능 매체.

청구항 37.

제34항에 있어서,

상기 적어도 하나의 다른 문서 중 적어도 하나의 아웃링크 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 수단을 더 포함하는 컴퓨터 판독가능 매체.

청구항 38.

제33항에 있어서,

상기 지정 수단은, 상기 문서와 동일한 기호 호스트 이름을 갖는 웹 서버 상에 위치한 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 수단을 포함하는 컴퓨터 판독가능 매체.

청구항 39.

제33항에 있어서,

상기 지정 수단은, 상기 문서와 동일한 도메인 상에 위치한 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 수단을 포함하는 컴퓨터 판독가능 매체.

청구항 40.

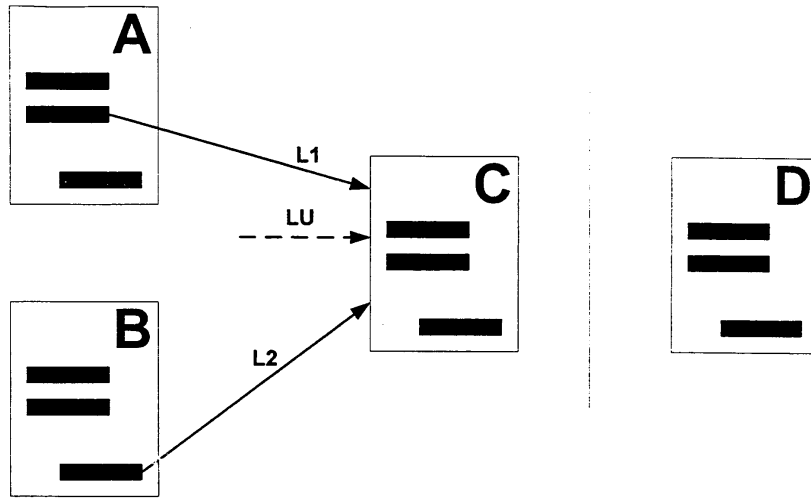
제33항에 있어서,

상기 지정 수단은, 상기 문서와 동일한 인터넷 프로토콜(IP) 주소와 관련된 문서의 수에 반비례하여 상기 문서에 상기 스코어를 지정하는 수단을 포함하는 컴퓨터 판독가능 매체.

도면

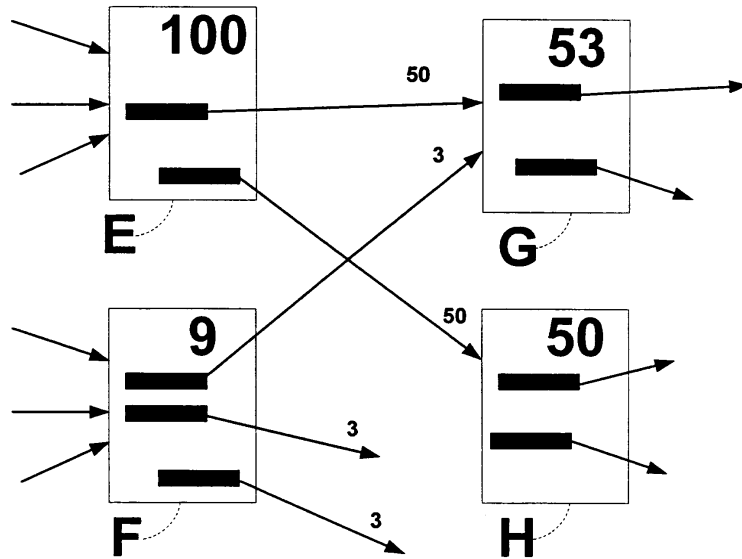
도면1a

(종래 기술)

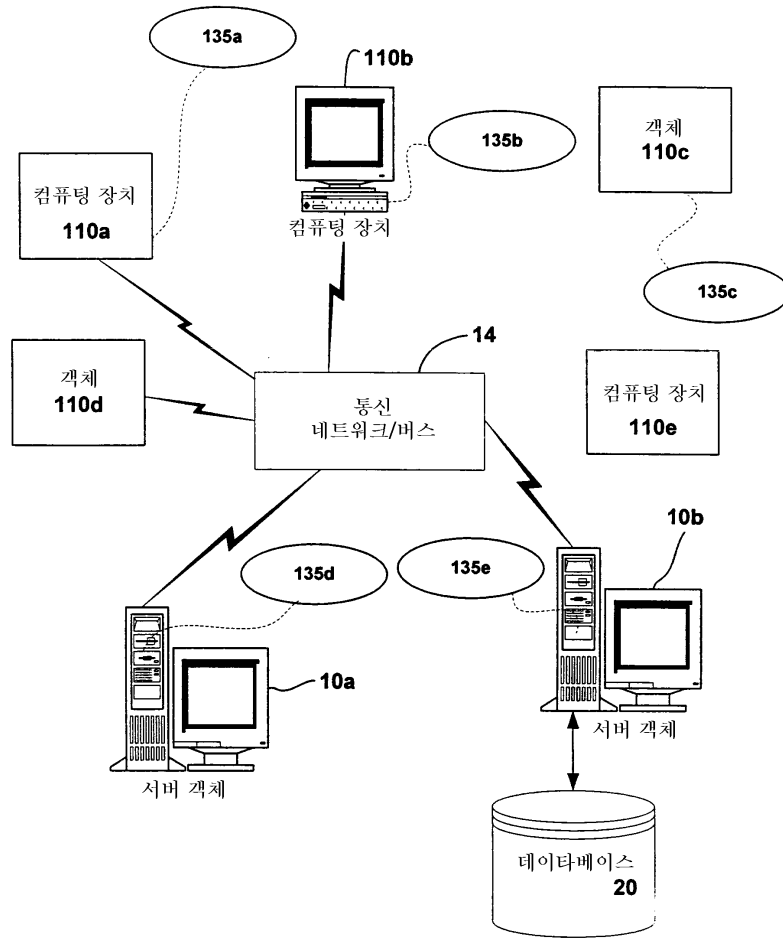


도면1b

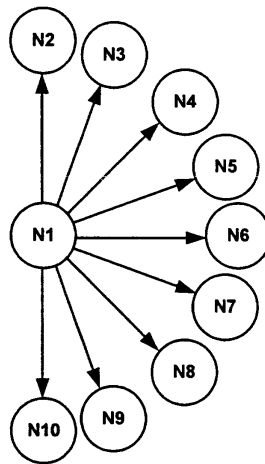
(종래 기술)



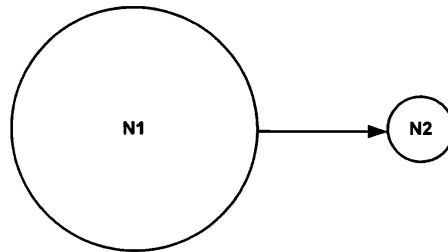
도면2a



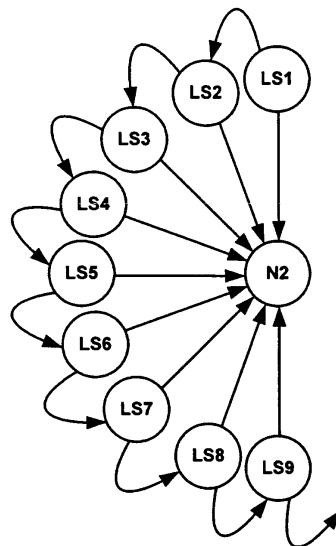
도면3b



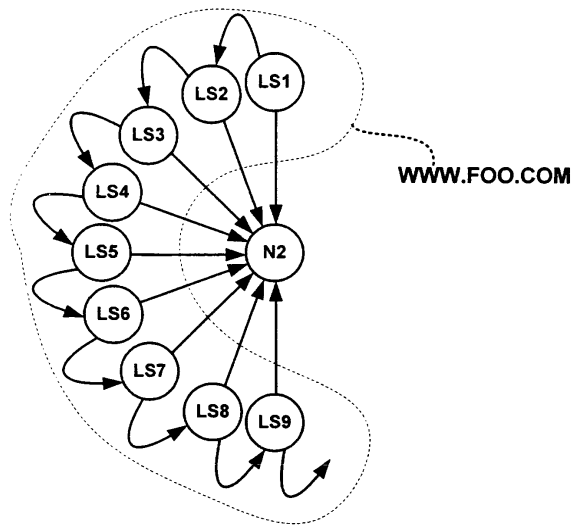
도면3c



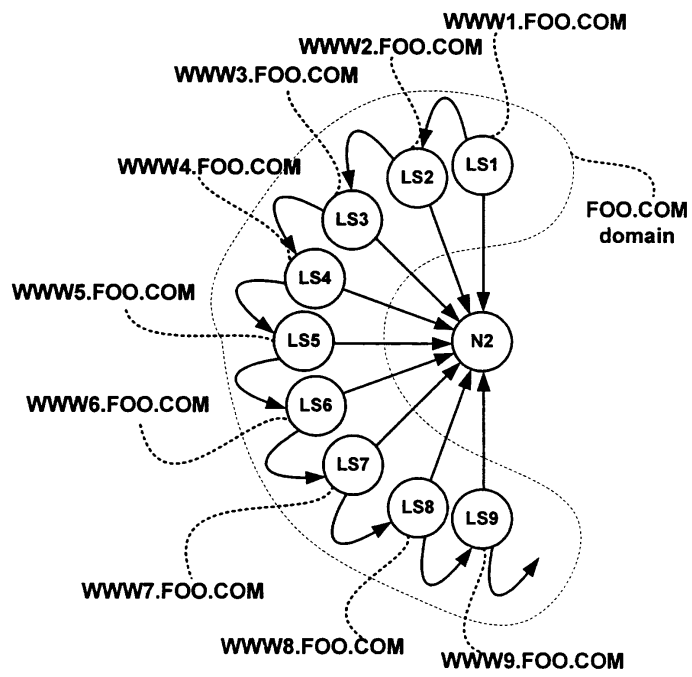
도면3d



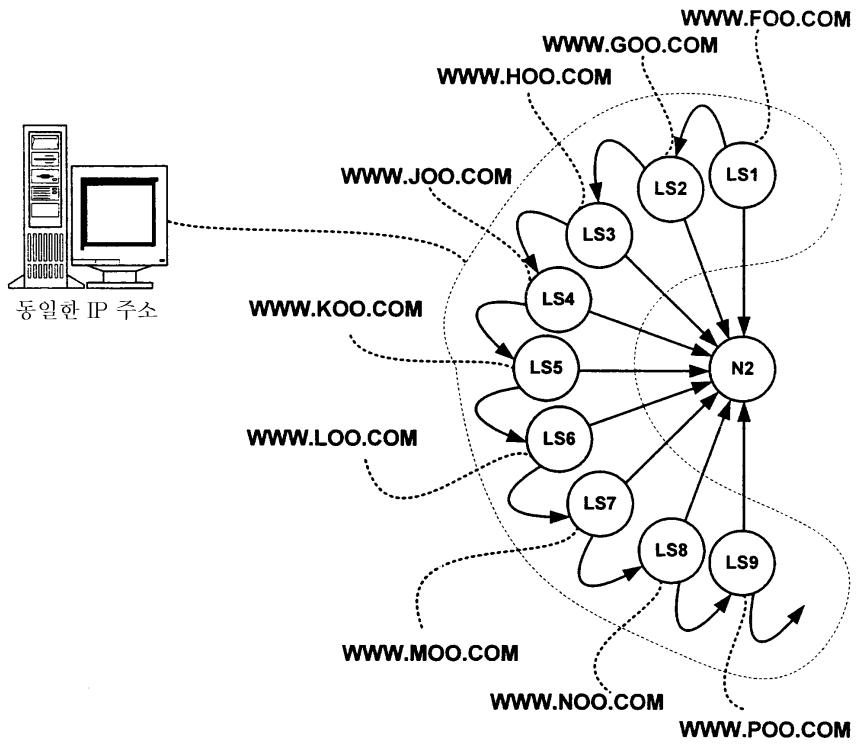
도면3e



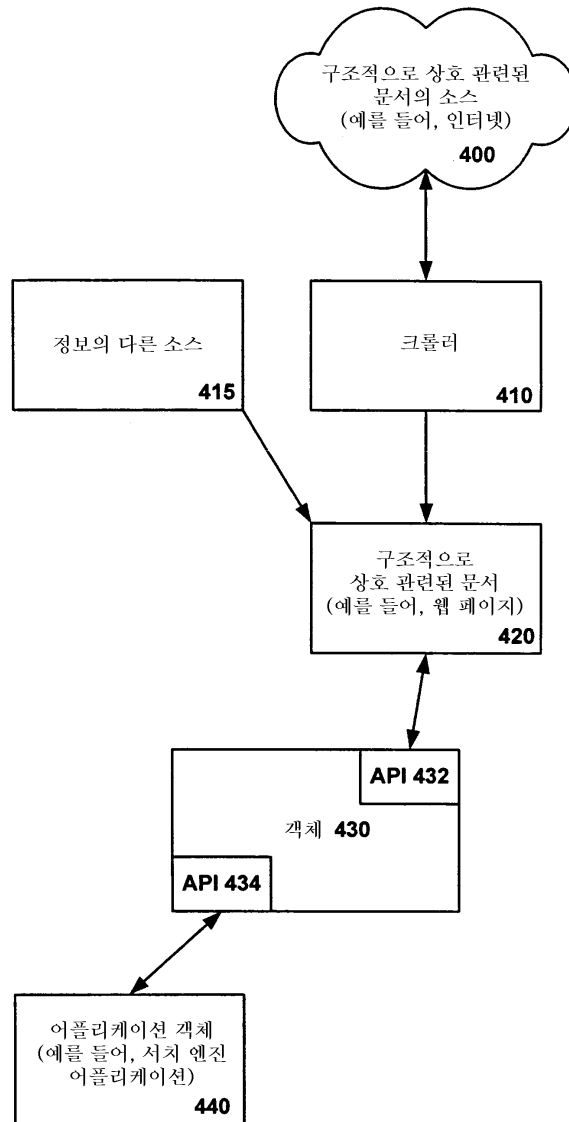
도면3f



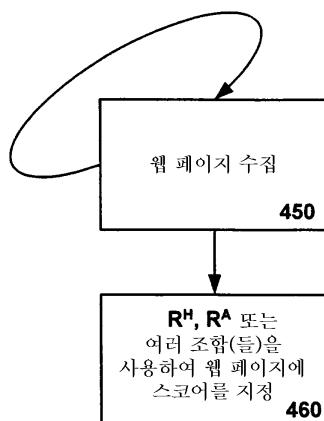
도면3g



도면4a



도면4b



도면4c

