



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2010-0022565
(43) 공개일자 2010년03월03일

(51) Int. Cl.

G06F 17/30 (2006.01)

(21) 출원번호 10-2008-0081135

(22) 출원일자 2008년08월20일

심사청구일자 2008년08월20일

(71) 출원인

충남대학교산학협력단

대전광역시 유성구 공동 220번지 충남대학교

(72) 발명자

김형식

대전 유성구 전민동 엑스포아파트 206-1306

유진형

대전 중구 선돌1길 59

(74) 대리인

김원준

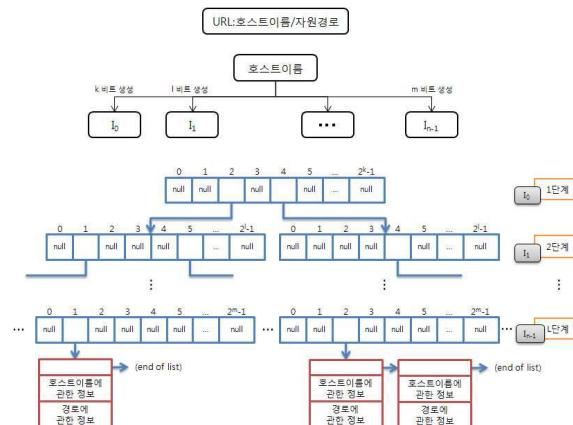
전체 청구항 수 : 총 5 항

(54) 해시트리를 이용한 URL 검색방법

(57) 요약

본 발명은 URL 목록에서 특정 URL을 검색하는 종래의 방법에서 검색 속도를 개선하기 위하여 해시트리를 이용한 URL 저장과 검색방법에 관한 것이다. URL 해시트리의 모든 노드(node)는 해시 테이블로 이루어지는데 내부 노드(internal node)의 해시 테이블은 하위 해시 테이블의 메모리 포인터를 저장하고 종단 노드(leaf Node)의 해시 테이블은 호스트이름에 관한 정보와 경로에 관한 정보를 저장하는 리스트의 메모리 포인터를 저장한다. 즉, URL 해시트리는 해시 테이블과 리스트로 구성된다. 본 발명의 목적은 URL 목록을 저장하고 있는 URL 해시트리를 이용하여 검색대상 URL을 빠르고 효율적으로 검색하는 것이다.

대표도 - 도2



이 발명을 지원한 국가연구개발사업

과제고유번호 C1090-0801-0016

부처명 지식경제부

연구사업명 대학 IT연구센터 육성 지원사업

연구과제명 해킹·바이러스 대응기술 연구

주관기관 충남대학교

연구기간 2003. 8. 1 ~ 2011. 12. 31

특허청구의 범위

청구항 1

- (A) (a) 사전에 정의된 URL 목록에서 해시트리에 삽입할 URL 하나를 추출하는 단계;
 (b) 추출된 URL을 호스트이름과 경로로 분리하는 단계;
 (c) 해시함수를 이용하여 호스트이름으로부터 해시값 튜플($I_0, I_1, \dots, I_{n-1}$)을 생성하는 단계;
 (d) 상기 해시값 튜플($I_0, I_1, \dots, I_{n-2}$)을 해시트리 각 단계의 인덱스로 활용하여 해당되는 중단 노드를 찾고 중단 노드가 존재 하지 않을 경우 내부 노드와 중단 노드를 생성하는 단계;
 (e) 상기 중단 노드의 인덱스(I_{n-1}) 위치에 호스트 이름에 관한 정보와 경로에 관한 정보를 리스트에 저장하는 단계;
 (f) URL 목록의 모든 URL에 대하여 상기 (a)~(e) 단계를 수행한 후 해시트리 생성과정을 완료하는 단계;를 포함하는 **해시트리 생성과정**:과
- (B) (a) 검색 대상 URL 입력단계;
 (b) 검색 대상 URL을 호스트이름과 경로 분리하는 단계;
 (c) 해시함수를 이용하여 호스트이름으로부터 해시값 튜플($J_0, J_1, \dots, J_{n-1}$)을 생성하는 단계;
 (d) 상기 해시값 튜플($J_0, J_1, \dots, J_{n-1}$)을 이용하여 상기 해시트리에서 리스트를 찾는 단계;
 (e) 리스트에서 저장된 호스트이름과 경로정보에 관련된 정보를 비교하는 단계;를 포함하는 **URL 검색과정**:으로 이루어지는 것을 특징으로 하는 해시트리를 이용한 URL 검색방법.

청구항 2

제 1 항에 있어서,

상기 과정(A)의 단계(e)에서, URL을 호스트이름 관련 정보와 경로 관련 정보로 구분하여 저장하는 것을 특징으로 하는 해시트리를 이용한 URL 검색방법.

청구항 3

제 1 항에 있어서,

상기 과정(A)의 단계(e)에서, URL을 리스트에 저장할 때에 경로에 관한 정보가 디렉토리일 경우 정보의 마지막에 '/'를 덧붙여 저장하는 것을 특징으로 하는 해시트리를 이용한 URL 검색방법.

청구항 4

제 1 항에 있어서,

상기 과정(A)의 단계(c)와 (d), 상기 과정(B)의 단계(c)와 (d)에서, URL의 해시함수를 이용하여 호스트이름으로 해시값 튜플을 생성하고, 각 해시값을 URL 해시트리의 각 단계 인덱스로 사용하는 것을 특징으로 하는 해시트리를 이용한 URL 검색방법.

청구항 5

제 1 항에 있어서,

상기 과정(B)에서, 검색대상 URL을 해시트리의 노드와 해시값 튜플을 이용하여 먼저 간략하게 검색하는 단계와 리스트를 이용하여 정확하게 검색하는 단계를 구분하여 검색하는 것을 특징으로 하는 해시트리를 이용한 URL 검색방법.

명세서

발명의 상세한 설명

기술 분야

[0001] 본 발명은 웹사이트 콘텐츠 필터링 시스템에 관한 것으로, 보다 상세하게는 URL 목록에 대한 해시트리를 생성하고 이를 활용하여 사용자에게 의해 입력된 URL이 미리 정의된 URL 목록에 포함되는지를 효과적으로 검색(필터링)하는 방법에 관한 것이다.

배경 기술

[0002] 최근 웹과 인터넷이 대중화 되면서 그 역기능이 문제가 되어 웹사이트 필터링 시스템의 필요성이 대두되었고, 웹사이트 콘텐츠를 효율적으로 필터링하기 위한 여러 방식들이 사용되고 있다. 또한 다양한 응용 프로그램들을 웹으로 서비스하려는 추세에 입각하여 웹 서비스 객체들이 URL(Uniform Resource Locator)로 표현되므로, 사용자의 요청 URL과 미리 정의된 URL 목록을 비교하여 웹사이트 콘텐츠를 필터링하는 방법(URL 목록 기반 필터링 방법)이 많이 사용되고 있다.

[0003] 한편 미리 정의된 URL 목록의 크기는 빠르게 증가하고 있다. 이러한 URL 목록의 크기 증가는 검색속도를 현저히 떨어뜨림으로써 응답시간을 증가시키고 이에 따라 네트워크 서비스 품질이 감소되는 현상을 유발하고 있다.

[0004] URL 검색방법에 있어서 URL 간의 포함관계를 정의하는 것이 중요하다. 본 발명에서는 URL을 호스트이름(hostname)과 경로(path)로 구분하고 경로는 다시 디렉토리(directory)와 파일(file)로 구성된다고 정의한다. 이때, URL A와 B의 호스트이름이 같고 A의 경로가 파일의 계층구조상 B를 포함할 때 'URL A는 URL B를 포함한다.'라고 정의한다. 예를 들어 URL A는 "www.hostname/directory1/" 이고 URL B는 "www.hostname/directory1/file2" 이라면 URL A는 URL B를 포함한다.

[0005] 이러한 포함관계를 검색할 수 있는 종래의 방법으로 단일 해시 테이블(hash table) 방식이 알려져 있다. 단일 해시 테이블 방식에 의하면, URL 목록을 정의하고 URL 목록에 존재하는 URL의 개수에 따라 적당한 크기의 해시 테이블을 생성한 후 해시함수를 통해 각 URL들의 해시 테이블 위치를 결정하여 URL들을 저장한다. 이러한 단일 해시 테이블 방식은 다음과 같은 두 가지 속성에 따라 검색시간이 길어지는 문제점이 있다.

[0006] 단일 해시 테이블 방식에서는 URL의 포함관계를 찾기 위하여 검색대상 URL의 경로를 분리하여 호스트이름에 하나씩 덧붙여가며 비교한다. 즉, 분리한 경로를 순차적으로 하나씩 호스트이름과 연결하고 해시 테이블을 참조하여 비교한다. 예를 들어 도1과 같은 URL 목록과 단일 해시 테이블에 대하여, 검색대상 URL이 "www.hostname.com/dir2/a.html"이라 가정하자. 이 방식에 의하면, 먼저 "www.hostname.com/"을 해시함수를 통하여 인덱스를 구하고 비교한다. 그러나 해시 테이블에서 해당 URL을 찾지 못하기 때문에 "dir2/"를 덧붙인 "www.hostname.com/dir2/"를 해시 테이블에서 검색하고 성공한다. 만일 여기서도 검색을 실패한다면 "a.html"도 덧붙여 계속 검색한다. 따라서 검색에 실패하거나 검색대상 URL의 경로가 많은 개수의 디렉토리로 구성되어 있다면 각 디렉토리를 호스트 이름에 덧붙여 비교해야하기 때문에 단일 해시 테이블 방식의 검색 시간이 길어지는 문제가 있다. 즉, 종래기술에 의한 URL 검색방법에 의하면, 검색대상 URL을 검색하기 위해서는 디렉토리들과 파일을 하나씩 호스트이름에 연결하고 해시함수를 이용하여 검색하는 단계를 반복하여 모든 디렉토리들과 파일을 덧붙여서 검색해야만 한다. 따라서 검색대상 URL을 검색하지 못한다면 디렉토리의 개수만큼 비교횟수가 증가할 수밖에 없다. 일반적으로 URL 검색응용에서 검색에 실패하는 비율이 성공하는 비율보다 훨씬 크기 때문에 종래 기술은 효율적이지 못한 검색방법이다.

[0007] 두 번째는 인덱스의 충돌 문제이다. 해시기법에 의하면 인덱스의 충돌 문제가 발생할 가능성이 있다. 해시 테이블을 구성할 때 해시 인덱스의 충돌이 발생하면 단일 해시 테이블은 다음 인덱스(slot) 위치에 저장하는 open linear addressing을 사용한다. 따라서 검색대상 URL을 해시 테이블에서 검색할 때는 해당 인덱스뿐만 아니라 해시 테이블에 저장된 URL이 NULL이 될 때까지 다음 인덱스와 계속 비교하여 찾아야 하므로 해시 테이블이 충분히 크지 않거나 해시 값의 충돌이 많이 일어날 경우 검색 시간이 길어지는 문제가 있다. 충돌을 줄이기 위해서 인덱스를 크게 한다면 해시 테이블의 크기를 키워야 하고 사용하지 않는 메모리 공간이 늘어난다. 이 문제는 단일 해시 테이블 구조에서는 저장하는 엔트리 수와 상관없이 인덱스의 크기에 따라서 해시 테이블의 메모리 사용량이 결정되기 때문이다.

발명의 내용

해결 하고자하는 과제

[0008] 본 발명은 전술한 종래기술의 단점을 해결하기 위한 것으로, 사전에 정의된 URL 목록으로부터 해시트리를 생성하고, 이를 이용하여 검색대상 URL을 빠르고 효과적으로 검색할 수 있는 방법을 제공하는 것을 과제로 한다.

과제 해결수단

[0009] 해시 테이블을 이용한 검색방법에 있어서 해시값의 범위가 클수록 충돌을 회피할 가능성이 높아져 검색 시간은 짧아지지만 메모리 공간이 제한되므로 해시값의 범위를 증가시킬 수 없다. 또한 이상적인 해시함수가 발생시킨 해시값은 고루 분포하지만 실제 해시함수가 생성하는 해시값은 집중되기 마련이고, 해시 테이블 전체 엔트리의 수가 삽입된 엔트리보다 큰 경우에도 충돌은 여전히 발생할 수 있다. 본 발명은 해시값의 범위는 키우되 사용하지 않는 엔트리의 메모리 사용을 줄이는 방법을 고안한다. 크기가 큰 단일 해시 테이블 구조가 아닌 비교적 작은 크기의 해시 테이블을 계층적으로 사용하고 엔트리가 삽입될 때마다 필요한 해시 테이블만을 할당함으로써 메모리를 효과적으로 사용할 수 있다. 예를 들어 24비트의 해시값이 단일 해시 테이블로 구성된다면 2^{24} 개의 모든 엔트리가 처음부터 할당되어야 한다. 그렇지만 3단계 해시트리로 구성하고 24비트의 해시값을 8비트 해시값 세 개로 분할하면, 단일 해시 테이블과 마찬가지로 2^{24} 개의 엔트리를 허용하면서도 할당되는 메모리 공간의 크기를 감소시킬 수 있기 때문에 결과적으로 동일 크기의 메모리를 사용하더라도 해시값의 범위를 키울 수 있다.

[0010] 본 발명은 이러한 원리를 활용한 것이다.

[0011] 전술한 목적을 달성하기 위한 본 발명은, 사전에 정의된 URL 목록으로부터 해시트리를 점진적으로 생성하는 과정(해시트리 생성과정, A)과, 상기 해시트리를 이용하여 검색대상 URL을 빠르고 효과적으로 검색하는 과정(URL 검색과정, B)으로 이루어진 URL 검색방법에 관한 것이다.

[0012] 본 발명에서 상기 해시트리 생성과정(A)는, (a) 사전에 정의된 URL 목록에서 해시트리에 삽입할 URL 하나를 추출하는 단계; (b) 추출된 URL을 호스트이름과 경로로 분리하는 단계; (c) 해시함수를 이용하여 호스트이름으로부터 해시값 튜플($I_0, I_1, \dots, I_{n-1}$)을 생성하는 단계; (d) 상기 해시값 튜플($I_0, I_1, \dots, I_{n-2}$)을 해시트리 각 단계의 인덱스로 활용하여 해당되는 종단 노드를 찾고 종단 노드가 존재 하지 않을 경우 내부 노드와 종단 노드를 생성하는 단계; (e) 상기 종단 노드의 인덱스(I_{n-1}) 위치에 호스트 이름에 관한 정보와 경로에 관한 정보를 리스트로 저장하는 단계; (f) URL 목록의 모든 URL에 대하여 상기 (a)~(e) 단계를 수행한 후 해시트리 생성과정을 완료하는 단계;를 포함한다.

[0013] 본 발명에서 해시트리를 이용하여 검색대상 URL을 검색하는 검색과정(B)는, (a) 검색대상 URL 입력단계; (b) 상기 검색대상 URL을 호스트이름과 경로로 분리하는 단계; (c) 상기 해시트리 생성과정에서 이용된 해시함수와 동일한 해시함수를 이용하여 호스트이름으로부터 해시값 튜플($J_0, J_1, \dots, J_{n-1}$)을 생성하는 단계; (d) 해시값 튜플($J_0, J_1, \dots, J_{n-1}$)을 각 단계 인덱스로 활용하여 상기 해시트리에서 해당 리스트를 찾는 단계; (e) 상기 리스트에서 저장된 호스트이름과 경로정보에 관련된 정보를 비교하는 단계;를 포함하는 것을 특징으로 하는 해시트리 이용한 URL 검색방법에 관한 것이다.

효과

[0014] 이상과 같이 URL 해시트리를 이용한 검색방법은 해시트리 노드와 인덱스를 이용하여 해당되는 리스트의 존재여부를 확인하고 리스트가 존재할 경우만 리스트를 이용하여 다시 검색하기 때문에 주어진 URL이 목록에 포함되어 있지 않은 경우 빨리 알아낼 수 있다.

[0015] 또한 본 발명에 의하면, 해시트리에서 사용하지 않는 인덱스의 노드는 할당하지 않기 때문에 같은 크기의 인덱스를 가지는 단일 해시 테이블 방법에 비하여 적은 메모리를 사용한다. 단일 해시 테이블 방법은 인덱스의 충돌을 예방하기 위하여 충분히 큰 해시 테이블을 이용하면 사용하지 않는 인덱스의 해시 테이블도 할당하는 단점을 가지고 있지만 해시트리를 이용할 경우 사용하는 인덱스의 해시 테이블만을 생성하기 때문에 실제 메모리는 사용하지 않고 가상의 인덱스 공간들을 얻을 수 있는 장점이 있다.

발명의 실시를 위한 구체적인 내용

[0016] 이하 도면을 참조하여 본 발명을 상세하게 설명한다. 그러나 이들은 예시적인 목적일 뿐 본 발명이 이에 한정되

는 것은 아니다.

- [0017] 도 2는 URL 해시트리의 일반적인 구조를 그린 도면이다. 해시함수를 이용하여 목록상 URL의 호스트이름으로부터 계산된 해시값의 튜플을 생성하여 해시트리의 인덱스($I_0 \sim I_{n-1}$)로 사용한다.
- [0018] 호스트이름으로부터 해시값의 튜플을 생성하는 방법은 여러 가지가 있다. 예를 들어 다중의 해시함수를 사용하여 여러 해시값을 얻을 수도 있고, 단일 해시함수로 얻은 단일 해시값의 부분을 취하여 여러 해시값을 얻을 수도 있다. 해시값 튜플에서 해시값의 개수는 해시트리의 총 단계수와 같고 해시값의 비트수는 해당 단계의 해시 테이블 크기와 관련이 있다.
- [0019] 해시트리의 내부 노드는 하위 단계의 해시 테이블의 메모리 포인터가 저장되고 해시트리의 종단 노드에는 호스트 이름에 관한 정보와 경로에 관한 정보를 저장하는 리스트의 메모리 포인터가 저장된다.
- [0020] 서로 다른 URL이 같은 해시값 튜플을 갖는 경우에는 리스트를 이용하여 여러 URL이 같은 종단 노드에 저장될 수 있도록 한다.
- [0021] 도 3은 URL을 입력받아 해시트리를 생성하는 과정을 순서도로 나타낸 도면이다. URL 해시트리를 생성하기 위하여 URL 목록을 입력 받고 입력 받은 URL 목록의 URL들을 다음과 같은 단계에 따라 점진적으로 해시트리에 삽입한다: (a) 삽입할 URL을 추출하고 (b) 호스트이름과 경로로 분리하고 (c) 해시함수를 이용하여 분리된 호스트이름으로 해시값 튜플을 생성한다. (d) 해시값 튜플을 해시트리 각 단계의 인덱스로 활용하여 해당되는 종단 노드를 찾고 종단 노드가 존재하지 않을 경우 내부 노드와 종단 노드를 생성한다. (e) 상기 종단 노드의 인덱스(I_{n-1}) 위치에 호스트이름에 관한 정보와 경로에 관한 정보를 리스트로 저장한다. 디렉토리를 저장하고 싶을 경우에는 디렉토리임을 따로 명시하지 않고, 디렉토리 이름에 '/'를 덧붙여 저장한다. 또한 상기 리스트는 추후 빠른 검색을 위하여 특정 값으로 정렬할 수 있다. (f) URL 목록의 모든 URL들을 대상으로 반복적으로 상기 과정을 실행하여 모든 URL을 해시트리에 저장한다. 이러한 과정을 거쳐 URL 목록의 모든 URL이 해시트리에 삽입된다.
- [0022] 도 4는 URL 목록을 이용하여 URL 해시트리를 생성하는 과정에서 앞 부분에서 3개의 URL을 저장하고 4번째 URL을 삽입을 나타내고 있는 도면이다. 호스트 이름이 같은 두 URL은 해시트리에서 같은 위치에 리스트로 저장되고 나머지 URL은 해시트리에서 다른 위치에 저장되었다.
- [0023] 도 5는 URL 목록을 저장하고 있는 해시트리에서 검색대상 URL을 검색하는 방법을 순서도로 표현한 도면이다: (a) 검색 대상 URL을 입력받아서 (b) 호스트이름과 경로로 분리하고 (c) 해시함수를 이용하여 호스트이름으로부터 해시값 튜플을 생성한다. (d) 생성된 해시값 튜플을 해시트리의 인덱스로 사용하여 리스트를 찾는다. 생성된 해시값 튜플로 리스트를 찾지 못했다면 검색에 실패하고 종료한다. (e) 리스트를 찾았다면 리스트를 검색하면서 저장된 호스트이름과 경로에 관련된 정보를 리스트의 마지막까지 비교한다. 만약 리스트를 생성할 때, 특정 정보를 이용하여 리스트를 정렬하였다면 노드의 끝까지 비교하지 않아도 된다. URL의 포함관계를 찾기 위하여 경로 비교 시 문자열 비교를 사용하고 문자열 비교 길이는 트리노드에 저장된 경로의 길이로 정한다. 따라서 트리 노드에 저장된 경로의 길이가 더 길 경우 문자열을 비교하지 않고 다음 노드로 이동할 수 있다.
- [0024] 도 6은 URL 트리를 이용하여 검색대상 URL을 검색하는 방법을 예시적으로 설명하고 있는 도면이다.
- [0025] "www.hostname.com/dir2/file.html"을 입력 받아 호스트이름과 경로로 분리하고 해시함수를 이용하여 분리된 호스트이름으로 해시값 튜플을 생성하고 생성된 해시값 튜플 (4, 2,...2)을 인덱스로 사용하여 리스트를 찾는다. 해당 인덱스에 리스트가 존재하므로 리스트의 첫 번째 노드와 호스트이름과 경로관련 정보를 비교한다. 호스트이름 관련 정보는 같지만 경로의 길이가 더 길기 때문에 리스트의 다음 노드로 이동한다. 해당 노드에서는 호스트이름 관련 정보가 같고 노드에 저장된 경로가 검색대상 URL의 경로를 포함하고 있기 때문에 검색에 성공한다.

도면의 간단한 설명

- [0026] 도 1은 종래 기술에 의해서 URL 목록에서 해시 테이블로 변환되는 일례를 보여주는 개념도.
- [0027] 도 2는 본 발명의 URL 해시트리의 일반적인 구조를 나타내고 있는 개념도.
- [0028] 도 3은 본 발명에 의해서 URL 목록에서 해시트리로 변환되는 과정을 설명하는 순서도.
- [0029] 도 4는 본 발명에 의해서 URL 목록에서 URL 해시트리로 변환되는 과정의 일례를 보여주는 개념도.

[0030] 도 5는 URL 해시트리에서 검색대상 URL을 검색하는 과정을 설명하는 순서도.

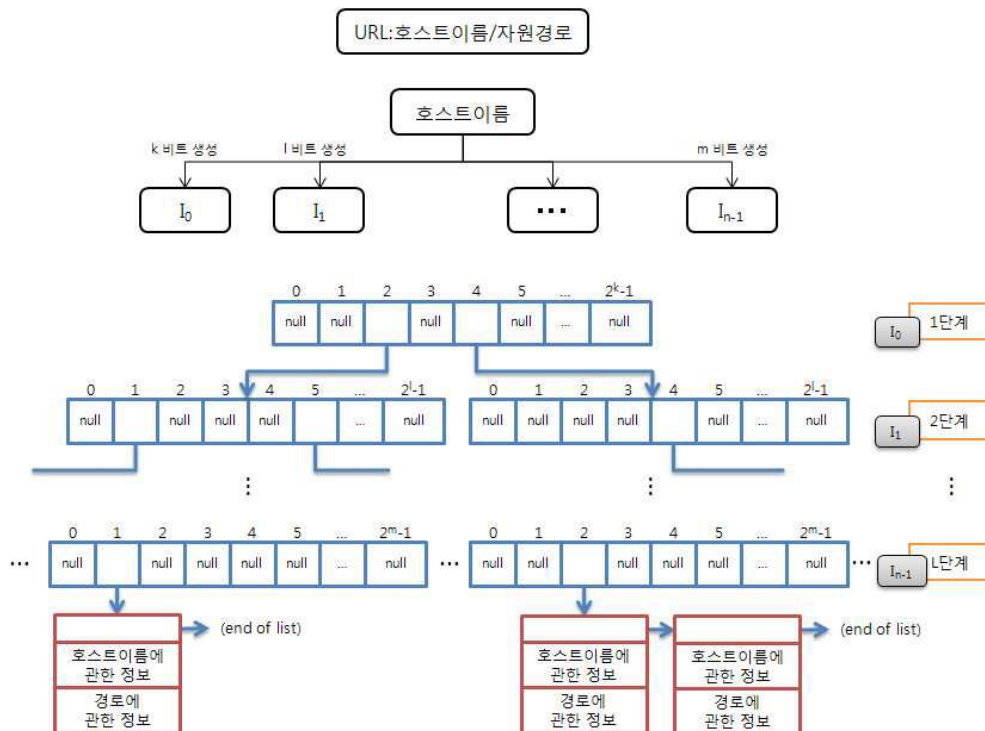
[0031] 도 6은 URL 해시트리에서 검색대상 URL을 검색하는 과정의 일례를 보여주는 개념도.

도면

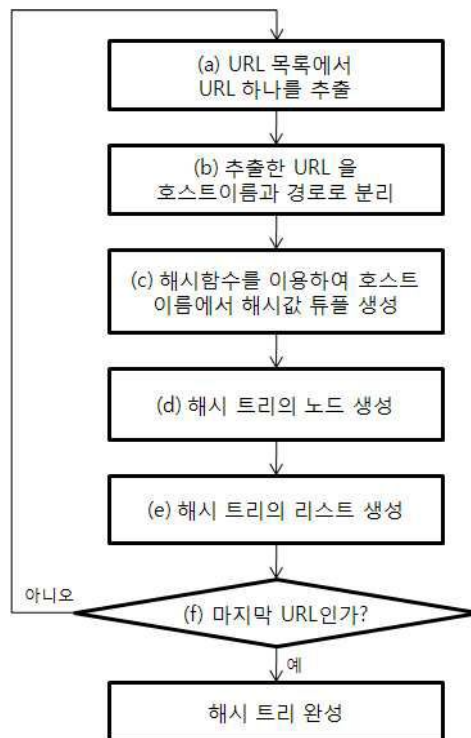
도면1



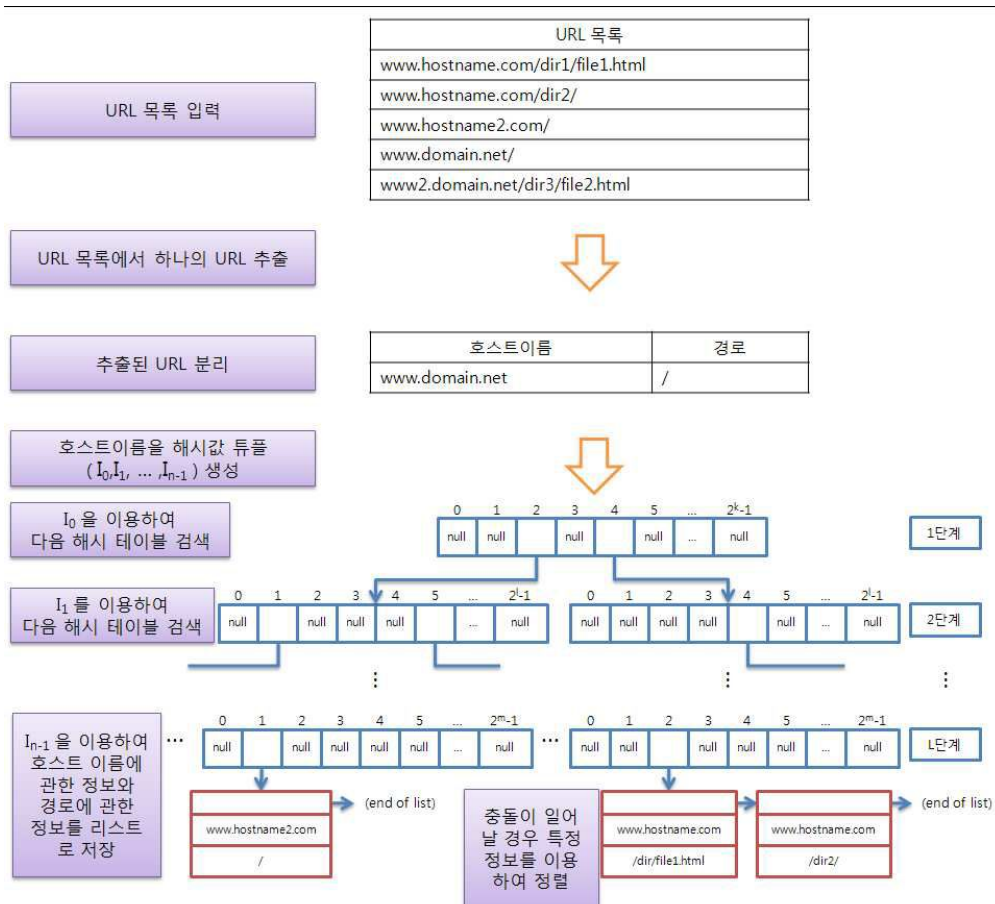
도면2



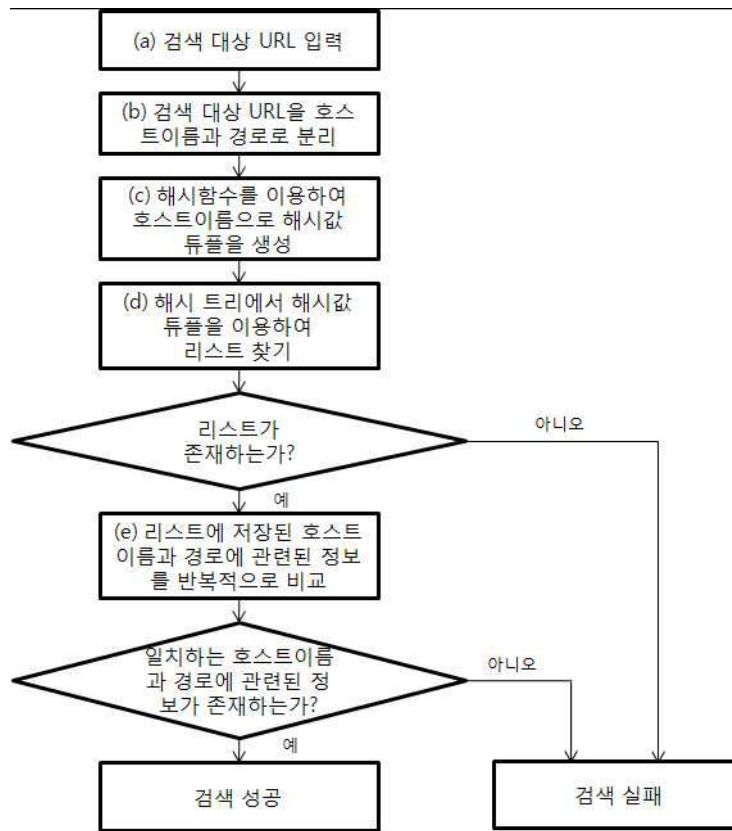
도면3



도면4



도면5



도면6

