**RDF-3X**

RDF-3X는 하나의 트리플에 6개의 순열 트리플을 B+tree로 구성되어 저장된다. 정밀한 선택측정을 위해 6개의 바이너리(즉, SP, SO, PS, OS, OP) 및 3개의 단일 예측을 마찬가지로 구성하여 B+tree를 유지한다. 인덱스들을 구성하기 전에 각각의 주어(Subject), 목적어(Object), 서술어(Predicate) 문자열이 int형 식별자로 매핑 되고 전역사전 안에 저장된다. 각 query 실행 계획은 접근을 기반으로 한 동적 프로그래밍을 기초로 최적의 비용을 사용하여 발생되어진다. 이는 트리의 내부의 SIP(Sideways Information Passing)라는 기술에 의해 더 최적화 될 수 있다.

RDF-3X는 RDF을 지원하기 위해 RISC의 “명령어 세트를 줄이자“라는 개념을 따르고 있다. 기본적으로 아래의 3가지 원칙을 따른다.

(1) 물리적 디자인은 하나의 큰 트리플 테이블를 적절히 인덱스하면서 작업량에 독립적이다.

RDF-3X는 자동 튜닝 마법사의 성공 또는 제한에 의존하는 것이 아니라, 물리적 디자인 튜닝의 필요성을 효과적으로 제거한다. 이는 3 차원의 6 가지 순열 전체에 대한 색인을 작성함으로써 가능하다. RDF 트리플을 구성하고 추가적으로 3 개의 2 차원 및 3 개의 모든 1 차원을 투영한 것에 대해 집계에 대한 인덱스를 생성한다. 모든 인덱스의 총 저장 공간은 기본 데이터의 크기보다 작다. 다시 말하면이러한 인덱스들은 잘 압축될 수 있다.

(2) 쿼리 프로세서는 주로 정렬된 인덱스 목록보단 병합조인에 의존한다.

이것은 트리플 테이블의 완전한 인덱스에 의해 가능하다. 즉, 모든 처리는 인덱스를 이용하고 트리플 테이블은 가상적일 뿐이다. 연산자 트리는 가능한 최대 범위까지 후속 조인에 대한 순서를 유지하도록 구성되는데 이것이 더 이상 가능하지 않을 때만 RDF-3X는 해시 기반 조인 처리로 전환한다. 이 접근법은 코드 수준에서 고도로 최적화 될 수 있으며 기존 쿼리 프로세서보다 훨씬 낮은 오버 헤드를 제공하고 이와 동시에 SPARQL의 다양한 중복 제거와 쿼리의 분리 패턴 및 SPARQL에 필요한 기타 모든 기능을 지원한다.

(3) 쿼리 최적화 프로그램은 주로 실행 계획 생성 시 조인 순서에 중점을 둔다.

RDF 특정 통계적를 기반한 비용 모델을 사용하여 열거를 위한 동적 프로그래밍을 사용한다. 이 통계에는 잠재적 결합 패턴인 데이터 그래프 경로의 빈번한 술어-시퀀스의 카운터가 포함된다. 범용 데이터베이스 시스템의 쿼리 최적화와 비교할 때 RDF-3X 최적화는 더 간단하지만 실행 계획에 대한 선택성 평가 및 결정은 오히려 훨씬 정확하다.

RDF-3X는 모든 트리플을 압축 된 클러스터 된 B+Tree에 저장한다. 트리플은 SPARQL 패턴을 범위 스캔으로 변환 할 수 있는 B+Tree에서 사전적으로 정렬된다. 패턴(literal1, literal2, ?x)에서 단어들은 공통 접두사를 지정하므로 효과적으로 범위 스캔을 지정한다. 중첩 할 수 있는 리프 페이지 수에 대한 단일 스캔 중에 ?x의 가능한 각 바인딩이 발견된다. 트리플에는 긴 문자열 리터럴이 포함될 수 있으므로 매핑 사전을 사용하여 모든 리터럴을 ids로 대체하는 방식을 채택한다. 이 것은 두 가지 장점이 있는데, 첫 째는 트리플 스토어를 압축하는데, id 트리플만을 포함한다는 점이고, 두 번째는 쿼리 프로세스의 단순화를 만든다. 쿼리 변환 중에 쿼리에서 발생하는 리터럴은 사전 ID로 변환되며 문자열에서 ID까지 표준 B+Tree를 사용하여 수행 할 수 있다. 쿼리를 처리 한 후에 결과 ID는 응용 프로그램 / 사용자에 대한 출력으로 다시 리터럴로 변환되어야한다.