XML 스트림에 대한 다중 연속 XPath 질의의 공유처리 기법

이현호[o1] 이원석[2]
[1]안양과학대학 컴퓨터정보학부
hhlee@ianyang.ac.kr
[2]연세대학교 컴퓨터과학과 데이베이스연구실
leewo@database.yonsei.ac.kr

Shared Evaluation of Multiple XPath Queries for an XML Stream

Hyun-Ho Lee[o1] Won-Suk Lee[2]
[1]Division of Computer Information, Anyang Technical College
[2]DB Lab., Department of Computer Science, Yonsei University

데이터 스트림은 빠른 속도로 발생되는 데이터 튜플 (tuple)들의 거대하고 무한한 시퀀스(sequence)로 정의된다. 데이터 스트림 관리시스템(DSMS)에 등록된 질의는 발생되는 튜플마다 수행되므로 연속 질의라 한다. 연속 질의는 실시간에 처리되어야 하며, 엄격한 시공간적 제약을 수반한다. XML 스트림은 XML 메시지나 패킷과 같은 논리적 단위(unit)들의 무한 집합으로 구성된다. 본 논문에서는 이를 청크(chunk)라 명한다. 청크는 관계형 데이터 스트림의 튜플과 같이 연속 XPtah 질의의 처리 단위이다.

많은 수의 연속 질의들이 DSMS에 미리 등록되기 때문에 이들의 공통 조건을 공유함으로써 집합적으로 이들을 처리하는 것이 보다 효율적이다. 이러한 목적으로 본 연구에서는 XPath 질의집합을 XP-table이라는 새로운 구조체로 변환한다. 어떠한 XPath 질의집합도 하나의 접두 트리(prefix tree)로 변환될 수 있다. 이러한 점에 착안하여, 주어진 XPath 질의집합은 접두 트리를 거쳐 XP-table로 변환된다. XP-table은 일련의 구성원 리스트(m-list)로 구성된다. 접두 트리의 루트(root)에서부터 비텍스트 단말(non-text leaf) 노드까지의 경로(path)를 기초 경로(base path)라 부르고, 기초 경로에서 변환된 일련의 조각요소(fragment)들을 조각 시퀀스(f-sequence)라 부른다. 조각요소란 XML의 단위 요소(element) 또는 속성(attribute)를 가리킨다. f-sequence의 인접한 두 조각요소는 반드시 부모-자식관계를 가져야 한다. f-sequence는 대응되는 기초 경로를 XML 구조 정보(DTD 또는 XML Schema)를 참조하여 조각요소들 간의 부모-자식 관계로 풀어냄으로써 얻어진다. XP-table의 m-list는 접두 트리의 기초 경로로부터 변환된 f-sequence 마다 만들어 진다. 기초 경로에 표현된 선택조건 (predicate)들을 근거로 대응되는 f-sequence의 전체 도메인(domain)은 일정한 수의 배타적 영역(region)으로 분할되며, m-list는 이러한 영역들에 대한 미리 계산된 매칭 결과를 유지한다. 주어진 XPath 질의집합은 XP-table이 가지는 여러 개의 m-list를 차례로 수행함으로써 집합적으로 처리된다. XML 스트림의 각 청크는 스트림 릴레이션(SR)이라 불리는 릴레이션 구조로 변환된다. SR은 청크를 구성하는 각 조각요소의 f-sequence와 관련 정보를 튜플로 저장한다. XP-table과 SR은 f-sequence를 매개로 서로 매칭된다.

본 연구의 공헌(contribution)은 다음과 같이 요약된다.

- 본 연구는 다중 연속 XPath 질의 처리를 위해, 그들의 선택조건을 물리적으로 공유하고 나아가 대상 질의집합의 부분적으로 미리 계산된 매칭 결과를 가지는 XP-table이라는 구체적 구조체를 제안한다.
- XP-table의 모든 구성요소는 실시간 질의처리 전에 모두 구축되므로, 실시간 부하를 최소화시키는데 기여한다.
- XP-table의 구성요소인 m-list의 영역 구조는 비교연산을 연산 종류에 관계없이 동일한 방식으로 처리하므로 비동등 연산도 동등 연산만큼 빠르게 처리할 수 있다.

그림 1은 제안된 시스템의 구성도이다. 제안된 시스템은 세 개의 하위 요소로 구성된다: 질의측 구성요소, 스트림측 구성요소 그리고 질의 수행기. 질의측 구성요소에서 연속 XPath 질의집합은 XP-tree로 컴파일되고, XP-tree는 다시 XP-table로 변환된다. 스트림측 구성요소에서는 SAX 파서가 XML 스트림의 각 청크를 스트림 릴레이션 (SR)로 변환환다. 질의 수행기는 XP-table의 m-list들을 차례로 SR과 매칭시키고, 매칭 결과를 매칭 연산 매트릭스 (MEM)에 저장한다. 각 XPath 질의의 최종 결과는 XP-expression을 연산함으로써 얻어진다. 질의측 구성요소는 모두 컴파일 시에 구축되는 반면, 스트림측의 SR은 실시간에 만들어진다.

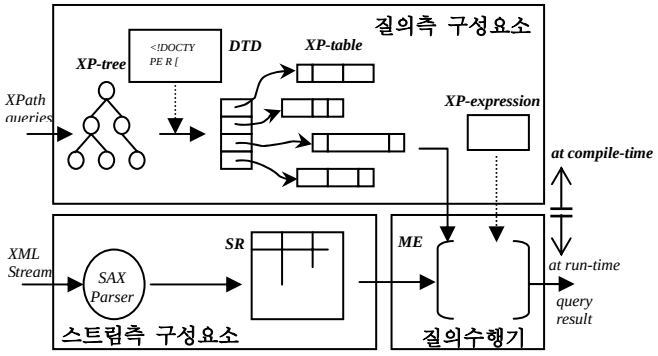


그림 1. 시스템 구성도

[1]연세대학교 컴퓨터과학과 박사과정. [1,2]정보과학회 정회원.
이 논문은 2007년도 정부(과학기술부)의 재원으로 한국과학재단의 국가지정연구실사업으로 수행된 연구임.
(No. M10600000225-06J0000-22510)

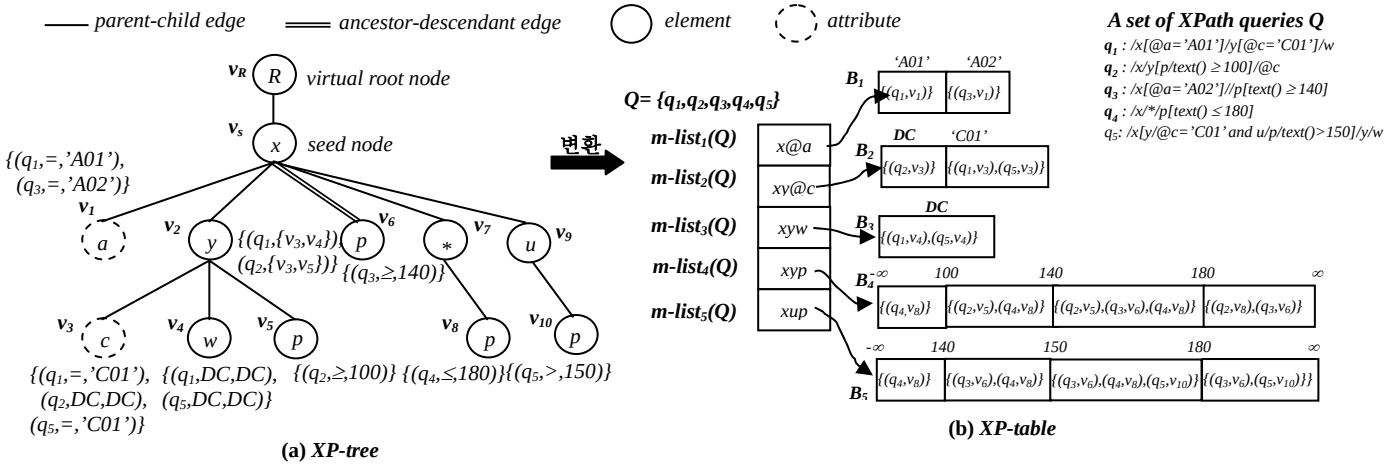


그림 2. 다섯 XPath 질의들에 대한 XP-tree와 XP-table

그림 2는 다섯 개의 XPath 질의에 대한 XP-tree와 이를 변환한 XP-table의 예를 보여준다. XP-table 구축 과정에는 XP-tree의 기초 경로에 대한 f-sequence를 구하기 위해서 목적 XML 스트림의 DTD를 참조한다. 주어진 질의 집합에 대한 XP-table과 개별 질의에 대한 XP-expression은 다음과 같은 XML의 반구조적(semi-structural) 문제를 해결한다.

- **비결정적 구성요소 문제** XPath 질의의 자손-또는-본인 관계(//)나 와일드 카드(*)와 같은 비결정적 구성요소의 문제를 풀기 위한 여러 가지 방법이 제안되었으나, 어떠한 연구도 실시간 질의처리 전에 이 문제를 완전히 해결하지는 못했다.

  → XP-tree의 단말 노드에 대한 기초 경로를 DTD를 참조하여 대응되는 f-sequence로 변환된다. 기초 경로 상에 자손-또는-본인 관계나 와일드카드 노드가 존재한다면, 만족하는 모든 가능한 f-sequence들 찾기 위해 DTD를 깊이우선(depth-first) 방식으로 탐색한다. 반대로, 서로 다른 단말 노드의 기초 경로들이 동일한 조각요소를 가리킨다면, 하나의 f-sequence로 변환된다.

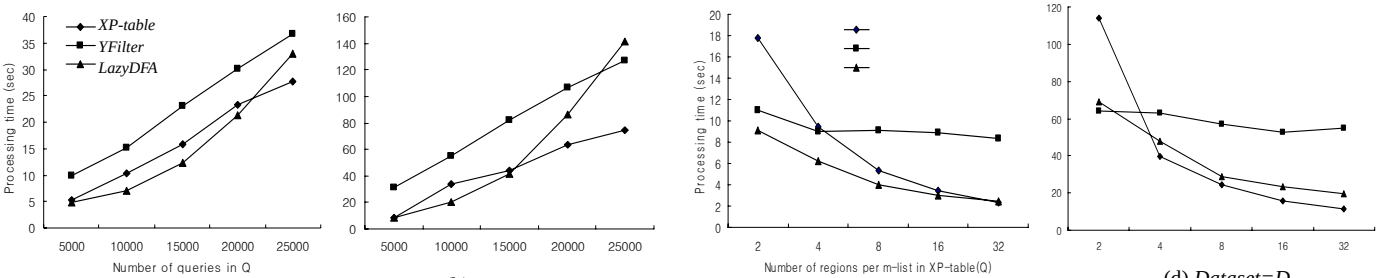
- **재귀적 조각요소 문제** 재귀적(recursive) DTD는 동일한 조각요소의 재귀적인 발생을 허용한다. 재귀적 조각요소를 포함한 f-sequence를 재귀적 f-sequence라 한다. 재귀적 DTD에서 제안된 접근 방법은 적절하지 못한 것처럼 보인다. 왜냐하면, 기초 경로에 존재하는 재귀적 조각요소는 무한대의 재귀적 f-sequence를 발생시키기 때문이다.

  → 재귀적 DTD에 대하여, 목적 XML 스트림의 최대 깊이와 같이 변환될 f-sequence 수를 제한하는 특정한 제약조건이 존재하지 않는다면, 본 연구에서는 최근 처리된 XML 스트림 청크 집합을 근거로, 변환될 f-sequence의 수를 한정함으로써 주어진 질의집합의 정확한 매칭 결과를 근접(approximation)한다. 한정 f-sequence 집합은 주기적으로 갱신된다.

- **비선형 XPath 질의 문제** XP-table의 m-list는 개개의 기초 경로들의 매칭 결과를 가지고 있기 때문에, 비선형 질의를 구성하는 둘 이상의 기초 경로들의 매칭 결과는 논리적으로 결합되어야 한다. 비선형 질의에서 부분적으로 공유되는 기초 경로들의 분리 지점 상의 요소를 가지치기 요소(branching element)라 부른다. 청크 내에 가지치기 요소가 두 번 이상 나타날 경우, 매칭된 공유 기초 경로들 상의 가지치기 요소들의 실체(instance)가 동일한 것인지를 판단해야 한다.

  → 질의 수행기는 대상 청크의 각 조각요소 실체에 version이라는 유일한 전역 일련번호를 부여한다. 비선형 질의에 대한 XP-expression은 해당 질의를 구성하는 기초 경로의 매칭 결과를 논리적으로 조합하는 것뿐만 아니라 version을 통한 가지치기 요소의 version 일치성도 파악한다.

본 연구는 YFilter나 LazyDFA와 같은 기존 방법론과의 비교를 포함한 일련의 실험들을 통해, 제안된 시스템이 질의 처리의 실시간 부하를 줄임으로써 시간 효율성이 중요한 스트림 환경에서의 안정적 데이터 처리 능력을 보여준다. 그림 3의 실험에는 다음과 같은 두 종류의 실제(real) 데이터셋이 사용되었다: 6MB Protein 데이터셋 D_1과 9MB NASA 데이터셋 D_2. 데이터셋 D_1은 비재귀적(non-recursive) DTD를 가졌으며, 최대 깊이는 *7*이다. 데이터셋 D_2는 재귀적 DTD를 가졌으며, 최대 깊이는 *8*이다.



(a) *Dataset=D₁* (b) *Dataset=D₂* (c) *Dataset=D₁* (d) *Dataset=D₂*

그림 3. 질의 처리시간에 대한 실험

제안된 시스템은 XQuery 처리, 스트림 조인 그리고 집단화 (aggregation)와 같은 좀 더 진전된 주제를 다루기 위한 기초를 제공한다. 제안된 시스템은 질의처리를 위해 XML 스트림을 튜플에 기초한 릴레이션으로 변환하기 때문에 조인과 집단화와 같은 관계형 연산 적용이 용이하다. 향후 연구는 이러한 연산을 적용하기 위한 시스템 확장에 초점을 맞출 것이다.