DB성능관리 툴 MAXGAUGE:
웹 버전 설계 및 구현

김동균1, 구호민1, 박학빈2, 이상원1
서울관대대학교1, ㈜EXEM2
{donggosarang0, hong3499}@hotmail.com, lbpark@ex-em.com, swlee@acm.org

DB Performance Management Tool MAXGAUGE:
Design and Implementation of its Web Version

Donggewn Kim0, Hongmin Ku, Larkbin Park, Sang-Won Lee
Sungkyunkwan University and EXEM Co. Ltd.

요 약

대량의 정보를 빠르게 처리하기 위해 DBMS의 성능관리는 필수적이고 이를 빠르게 처리하기 위해 많은 도구들이 있다. 이들 도구 중 하나인 캐산 Maxgaugex는 오라클 데이터베이스의 성능 현황이 어떤지 조사하고, 설정된 웹과 서버를 충족하는 기능을 제공한다. 하지만 이는 설치의 변경을 요구하지 않고 데이터베이스 관리자에게만 제공할 수 있다는 한계가 있기 때문에 우리는 웹을 통해서도 시간과 장소에 구애 받지 않고 성능을 관리할 수 있게끔 플라이언트/서버 버전을 코어 버전으로 설계 및 구현하였다.

1) 서론
2) MAXGAUGE 소개
3) Web버전 설계 및 구현
4) 결론 및 향후 연구방향
5) 참고문헌

1. 서 론

최근 대량의 정보에 대응한 온라인 웹서비스를 요구하는 기업들이 급속히 증가하고 있는 추세이며, 실제로 많은 성능저하 요인으로 인해 최적의 서비스를 제공하지 못하고 있는 실정이다. 이에 성능관리가 필요하다면 시스템의 문제점과 최악이 이루어지지 않다는 것을 관리할 수 없게 되는 한계가 발생한다. 그러나 DBA가 하루 24시간 항상 당기일에 실시간 및 일정일 때를 수 행할 수 없고 실시간 감시 중에 성능문제가 발생하더라도 DB의 재구성, 세션의 강제 종료 등의 어려움이 있다. 이러한 이유로 이러한 실시간조직을 구성하고 있는 DB, 시스템, 여러분이 개인

2. MAXGAUGE 소개

2.1 주요기능

실시간 감시: 복수 개의 오라클 인스턴스를 한 화면에서 모니터링함으로써 생성되어 있는 오라클 성능문제를 수시로 인지하고 대처할 수 있다. 또한 하나의 서버에 대해 여러 인스턴스를 비교해보면서 모니터링할 수 있다.

2.2 시스템 간단: 하나의 오라클 인스턴스에서 발생한 성능문제의 원인을 파악한 시스템을 마우스 클릭에 의해 유리하게 추적할 수 있을 뿐만 아니라 특정 서버에 대한 정보 및 다른 SQL 변경을 한 화면에서 유리하게 모니터링함으로써 세션의 성능현황을 실시간에 파악할 수 있다. 또한 인스턴스에 접속하는 여러 개인의 세션들에 대한 정보를 하위 장에 의해 관리하고 조건에 맞는 세션 집합들을 동시에 검색한다. 하나의 세션에 대해서는 특정 정보, 대기 정보, 험들 SQL 문장, 접속 정보 등을 조회할 수 있다.

2.3 사후분석 및 트:relative: 오라클 인스턴스에서 발생되는 Performance Statistics, Waiting Event 등 성능정보의 후 시점의 세션 정보와 SQL 강의를 제공함으로써 고유의 성능 문제를 정확하게 분석하고 조건에 따라 최적의 성능을 보장한다.

그 외 Lock(LOCK) 추적, 병렬수행(PO) 추적, 원리 세션 추적, 지표별 세션 추적, SQL 검색, SQL 루프 조합, 오프젝트 조합, DBA 지원 등의 기능을 제공한다.

2.2 구조

구조는 그림 2와 같이 각각의 세부기능은 다음과 같다.

2.1.1 RTAD(Real-Time Access Daemon): DB 서버 내에서 수행되는 프로세스로서 SGA(Shared Global Area) 메모리를 직접 액세스하여 오라클의 성능정보를 수집하고 O/S kernel로부터 운영체계의 성능정보를 실시간 수집하는 프로세스다.

2.2.1 LOGD(Logging Daemon): DB 서버 내에서 수행되는 프로세스로서 RTAD에 동일한 방식으로 성능정보를 수집하여 수집된 정보를 파일에 기록한다.
3. MAXGAUGE 웹 버전

3.1 목적 및 가능
기존의 서버/클라이언트 방식은 회사내의 클라이언트 프로그램이 설치된 PC에 한하여 모니터링을 할 수 있었던 반면, 회사외부에서도 실시간 정보를 이용하여 모니터링을 할 수 있게 해보자는 취지에서 제작하였다. 주요 기능으로서는 앞서 소개한 MAXGAUGE에서 사후처리 및 통합을 제외한 모니터링에 영향을 두어 개발하였다.

3.2 구조
일반적인 Web환경을 구성하기 위해 다음의 구조를 생각한다. 그림 3과 같은 Unix서버에 jsp 언어를 사용하는 것과 그림 4의 IIS서버 기반에 Asp언어를 사용하는 것이다. 이 두 구조 중 관리측면, 비용측면 등의 요소를 저출할 할 수 있다. 서비스 대상이 소수 및 영업 점과 별도의 관리부서가 없는 점을 고려하여 기존의 window가 설치된 일반 PC환경에 IIS를 이용하여 서비스 하기로 했다. IIS기반의 자세한 구조는 다음과 같다.

3.3 설계 및 구현
본 시스템은 다음과 같은 설계원칙에 따라 설계하였다. 기본적인 구성으로서 서비스는 IIS를 이용하여 ASP.Net을 사용하고 클라이언트 측 사용자들에게 보여질 페이지는 Java Script, VB Script, Dhtml, Vml을 이용하여 보여지게 된다.

3.3.1 서버
RTAD와 직접 소켓을 통해 데이터를 주고 받는 형식으로 다음과 같이 구현된다.

3.3.1.1 서버측 브라우저 최소로 한다.
본래 3시간 간격으로 데이터를 요청하게 되는데 구조가 복잡하고 동작률이 떨어지면 문제가 발생할 수 있다. 따라서 RTAD로부터 받은 데이터를 바로 클라이언트로 넘겨주는 역할만 담당한다.

3.3.1.2 브라우저의 추가 구성 비용이 들지 않는다.
오라클이 설치된 서버와는 별도의 RTAD를 접속할 필요 없이 기존의 개발용 PC를 서버(IIS)로 활용한다.

3.3.1.3 C#기반의 ASP.NET 사용으로 재사용성 극대
소켓연결부분은 C#기반의 DL(Dynamic Link Library)을 이용해서 처리하므로 소스의 유지, 보수가 용이하다.

3.3.2 클라이언트
서버로부터 데이터를 받아 처리한 후 실시간으로 그래프의 지표를 브라우저에 보여 주는 역할을 한다. 실시간 그래프에서 가장 중요한 정밀도측면에서 최상의 결과를 얻기 위해 다음과 같은 방식을 제작하였다.

3.3.2.1 서버와의 접속횟수를 최소로 한다.
첫째, 한편 데이터의 전달에 걸리는 시간이 짧아져서 브라우저의 사용에 영향을 줄 수 있다고 판단하기에 시간을 최소화하였다.

3.3.2.2 Xml 기반의 GUI제공
웹 페이지에 그래프를 그리는 방법으로 크게 두 가지를 고려하였다. 첫째, 서버 측에서 그래프를 이미지로 만들어 클라이언트의 실시간으로 전달하는 방법, 둘째, script를 사용한 DOM을 사용하여 Xml기반의 웹 그라디 도구인 Vml(Vector
Markup Language)을 이용하는 방법이다. 기존에 쌍에서 사용하는 경우. 실제 같은 아이디 저장방식은 서버측에서
이들과 관련하여 전송시 부하가 상당히 커지고 또한 클라이언트측 이벤트로 서버측에 다시 접속을 유도하게
된다. 즉 전제적인 생성에서 임설한 측면이다.

3.3.2.3 javascript를 이용한 클라이언트측 모듈
ASP.NET에서는 크게 두가지 구분을 한다. 서버과정과 클라이언트과정이 그것이다. 처음에는 서버과정을 이용하여
RTAD로부터 받은 데이터를 저장하고 밑으로 표현하기 위해
Asp.net behind code(C#)에서 접속, 저장 및 전송을
담당하였으나 이는 Asp.net의 서버측에서 돌아가는 특징 때문에
알티유지 접속시 서버측에 상당한 부하를 줄여서 서버가
쾌속히는 문제를 발생시켰다. 또한 사용자 PC에 framework을
요구한다. 대신으로 클라이언트과정을 이용하여 사용자들에게
서버의 부담을 나누는 방식으로 클라이언트에서 동작하는
javascript를 이용하여 데이터를 가공하고 표현한다.

3.3.2.4 힌트(hidden) 프레임을 이용한 데이터 정신
또한 데이터를 그래프로 가공하여 화면에 부가분 따올리는
시점을 잡는다. 시작명의 결과 보통 3초의 시간마다 정신을
하나 데이터를 받아오는 시간을 1초로 하여 표현하기로 하는
시간은 2초라고 할 수 있을 정도다. 결국 3초마다 갱신되는 것이
아닌 좀 더 자연스러운 4~5초의 시간을 갱신하게 된다. 이러한 문제점
을 개선하기 위해 힌트 프레임을 들이. 바로 이곳에서 데이터를
전송받아 가공하여 저장하고 그래프에 부여해주기 위해 전단
계의 일을 수행한다. 나머지 나머지는 프레임용 단지 부여해주기
만 하는 역할을 함으로써 정신에 관리는 부하를 상당부분 감소
시킬수 있다.

3.3.2.5 cookie를 이용한 사용자 관리
사용자마다 달라 요구하는 지표를 cookie를 통해 관리한다.

3.4 구현
그림 5의 서버측에서 받은 데이터의 변화량을 계산하여 그
그래프를 통해 시각화하여 표현한 화면이고 그림 6은 각각의 지표
를 클릭했을시 특정 세션에 대해 현재 알람 및 메시. SQL 정보
를 나타내는 화면이다.

그림 6. Session Detail 화면

4. 결론 및 향후 구현 방향
4.1 결론
DB, 시스템, 애플리케이션 등 각각의 프로세스들에 대한 장
애 및 성능관리를 효과적으로 지원하기 위한 성능관리 도구인
MaxGauge를 주 5일 근무의 확산, VPN의 보급 등으로 인한 재
능력급가 중단화되고 또 출장업무가 많은 DBA에게 시간과 공
간제약 없이 모니터링할 수 있게 됨 범위로 개발하였다. 이 제품
중 MaxGauge와 함께 Package화되어 많은 DBA들에게 보다
편한 서비스를 제공하기를 기대한다.

4.2 향후 구현 방향
보다 상세한 SessionDetail(세션 추적기능, 세션 모니터링기
능 달달)화면 DHTML을 이용한 event 및 tree를 구현해야 하
고 현재 MaxGauge는 다중 RTAD와 통신이 가능하다. 현재 버전
은 현재의 구조대로라면 1개의 RTAD만 통신 가능하다. 이를 더
여서 말해 임시에 여러 개의 RTAD와 통신이 가능하게
물 구현하거나 또는 IIS 내부적으로 port를 달리하여 port별로
서로 다른 RTAD 관리하는 방안이 필요하다.

5. 참고문헌
정보통신학, 2002.

** Acknowledgement: 본 연구는 김동권, 구호민이 2004년
경을 만한 동안에 레이싱에 인접한 훈련으로 파견된 동안 이루어졌다. 이러한 기회를 제공해준 레이싱에 감사의사를 표한다.