

Toad 100배 활용하기

Toad for Oracle v.10
(제4편: 생산성과 효율성 향상을 위한 Toad Family 제품)

2010년 3월
Toad BDM, 김정중 부장
Jackson.kim@Quest.com



Toad for Oracle v.10 (제4편: 생산성과 효율성 향상을 위한 Toad Family 제품)

전 세계 50만 명의 매니아층을 보유한 데이터베이스 개발 툴의 대명사, Toad v.10 런칭을 기념하여 2009년 11월 18일 유저 세미나를 개최하였습니다. 퀘스트소프트웨어의 베스트셀러 Toad for Oracle 제품은 지난 10년간 전 세계 데이터베이스 개발자들에게 사랑을 받아온 최고의 데이터베이스 개발 및 관리 도구입니다.

자동화, 유저빌리티, 교육 및 유연성 등 진화의 진화를 거듭한 강력한 Toad의 새로운 기능을 소개하는 시간을 갖고자 합니다.

제4편은 고객들에게 성능, 튜닝, 부하테스트, 코드 테스트 등의 다양한 제품을 제공하는 Toad Family를 소개합니다. 어떤 담당자가 어떤 업무를 위하여 필요한 기능인지 간략한 소개로 Toad의 지식 및 Skill을 넓히는 기회를 제공합니다.

Toad for Oracle v.10의 업그레이드된 주요 기능은 다음과 같습니다.

1. 유니코드를 통한 완벽한 한글지원 및 다국어 동시지원
2. Toad for Data Modeler에 기반된 기술과 화려한 레포팅을 자랑하는 ER 다이어그램
3. 다중 데이터베이스 업무 생산성 향상을 위한 멀티 태스크 실행 기능
4. 튜닝 전문툴인 SQL Optimizer와 동일 인터페이스를 가진 토드의 SQL Tuning 기능
5. 데이터 조작의 시간절약과 효율성을 채택한 데이터 그리드 기능

Toad for Oracle v.10 (제4편: 생산성과 효율성 향상을 위한 Toad Family 제품)

Toad 활용을 위한 튜닝 기능은 무엇인가? – SQL Optimizer

서비스 제공을 위한 모든 애플리케이션 운영환경에서 데이터베이스의 성능은 서비스 품질을 좌우한다. 애플리케이션의 가장 중요한 요소로 SQL을 꼽을 수 있으며, SQL의 수행속도가 DB성능에 많은 영향을 끼친다. DB성능 향상을 위하여 SQL을 파악하고, 대안의 비교 및 검토하는 기능은 개발자와 DBA가 DB성능으로 다가갈 수 있는 직접적인 첫걸음이 된다.

- 1) ① 비교 가능한 대안 SQL을 실행비용 및 수행시간 등으로 비교하도록, SQL문을 한 눈에 정렬하여 보여준다.
- ② [Compare Scenarios] 메뉴를 클릭하면, 체크한 모든 시나리오의 비교 및 일대일 비교 기능을 제공한다.
- ③ 대안으로 선택된 [Alt#3]과 [Alt#2]를 통하여 상대방 SQL문의 차이점을 분명하게 표현한다.
- ④ [Table Access Full]에 고객테이블이 있는 경우, 칼럼 구성 및 성능정보를 팝업 윈도우로 확인한다.

The screenshot displays the SQL Optimizer interface in Toad for Oracle v.10. It shows a comparison of three scenarios (Alt #4, Alt #5, Alt #9) with identical results. The 'Compare Scenarios' menu is highlighted with a red box and a yellow circle labeled '2'. The 'Compare Scenarios' table is also highlighted with a red box and a yellow circle labeled '1'. The execution plan for Alt #3 is shown, with a 'TABLE ACCESS FULL' operation highlighted in yellow and a yellow circle labeled '3'. The execution plan for Alt #2 is also shown, with a 'TABLE ACCESS BY INDEX ROWID' operation highlighted in yellow and a yellow circle labeled '4'. A red box highlights the 'Columns' tab in the table definition window for 'QUEST_OPTI.CUSTOMER', with an orange arrow pointing to it from the 'TABLE ACCESS FULL' operation in the Alt #2 plan.

Scenario	Results Comparison	Plan Cost	Optimize: Total Elapsed Time	Total Elapsed ...	Logical Reads	Total CPU	First Row Elapsed
Alt #4	Identical	10		00:00:00.000	45	0.00	00:00:00.000
Alt #5	Identical	10		00:00:00.000	45	0.02	00:00:00.000
Alt #9	Identical	10		00:00:00.000	45	0.01	00:00:00.000

Toad for Oracle v.10 (제4편: 생산성과 효율성 향상을 위한 Toad Family 제품)

Toad 활용을 위한 SQL 부하테스트 기능은 무엇인가? – Benchmark Factory

DB 성능을 좌우하는 SQL문의 수행속도는 가장 중요한 튜닝요소 중의 하나이다. 더구나 서비스 운영자는 수 십 명에서 수 백 명에 이르는 사용자가 동시 수행할 때, 애플리케이션에서 발생한 SQL의 전체 응답시간이 중요하다. Benchmark Factory는 이러한 운영자의 요구사항에 맞추어 최상의 솔루션을 제공한다. 사용자 수에 따른 응답시간을 통하여 튜닝 이후 선별된 SQL 들을 최상의 대안으로 만들어 준다.

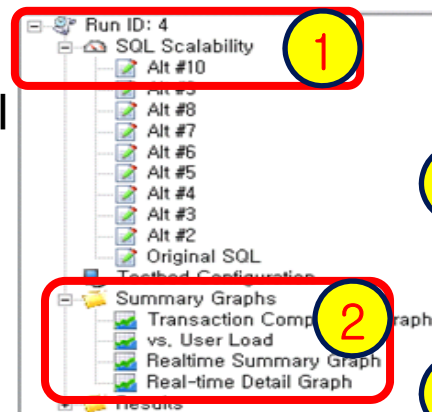
2) ① [SQL Scalability]는 선별된 SQL의 부하테스트를 위한 대안을 비교한다.

② [Summary Graphs]는 사용자의 증가에 따른 부하를 한 눈에 파악하도록 깊이있는 자료를 제공한다.

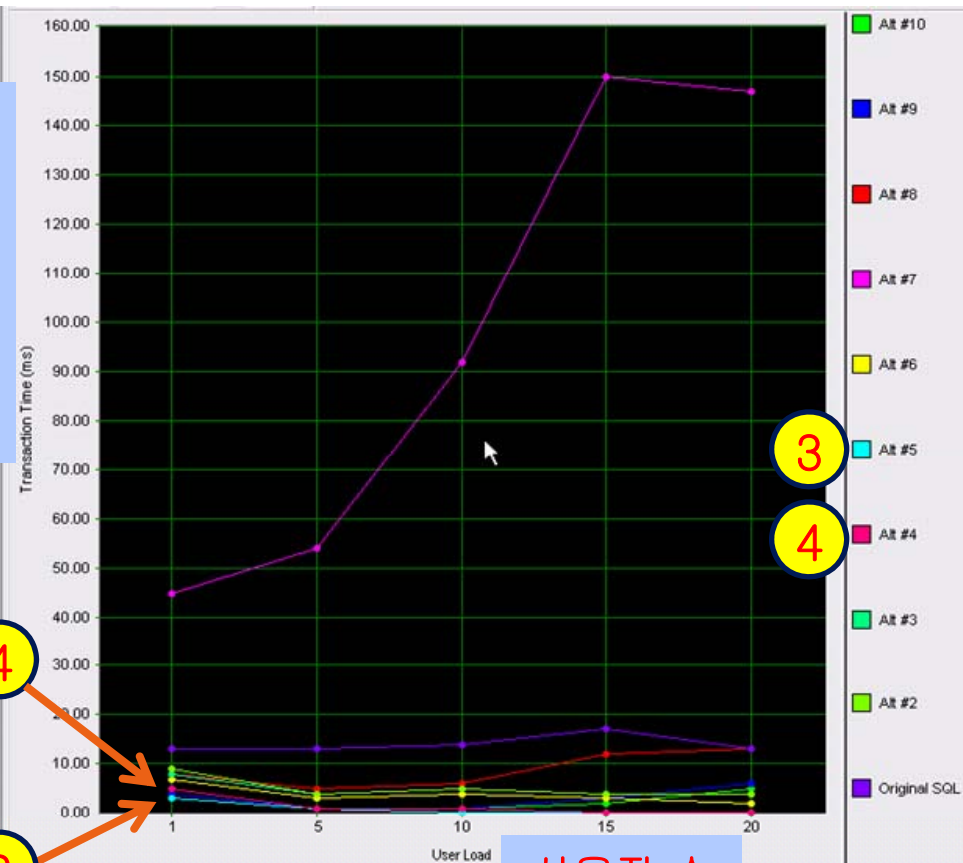
③ [Alt #5]는 1명의 사용자인 경우에 가장 낮은 트랜잭션 시간을 기록하지만, 20명일 경우에는 4,5 위로 밀려난다.

④ [Alt #4]는 1명의 사용자인 경우에 두 번째로 낮은 트랜잭션 시간을 기록하지만, 20명일 경우에는 1위로 올라간다.

향후 사용자 수 증가에 따른 최상의 SQL 선택이 DB성능에 중요한 역할을 한다.



트랜잭션 시간



Toad for Oracle v.10 (제4편: 생산성과 효율성 향상을 위한 Toad Family 제품)

Toad 활용을 위한 PL/SQL 테스트 케이스 환경은? – QCTO(Quest Code Tester for Oracle)

데이터베이스 애플리케이션을 구성하는 두 요소는 SQL과 PL/SQL이다. SQL의 수행과 분석은 Toad를 이용하면 효과적으로 사용할 수 있다. 그러나 PL/SQL의 경우에 이를 수행하고 결과를 확인하는 테스트 환경을 만들기는 그렇게 쉬운 문제가 아니다. QCTO는 PL/SQL의 코드 디버깅과 더불어 코드를 미리 수행하여 그 결과를 확인하는 시뮬레이션 기능을 제공한다. 테스트용 환경 구성시간을 극적으로 감소시킴으로써, 코드의 Test Case 생산성 향상에 많은 도움을 제공한다.

- 3) ① [Test Builder]는 3가지 스텝으로 PL/SQL 유닛을 위한 단순하고 복잡한 Test Case를 만드는 워크플로우를 제공한다.
- ② [Step 1], 유닛명과 테스트명을 입력하고, 유닛용 입력값을 넣는다.
- ③ [Step 2], 테스트 타입과 테스트 결과에 대한 기대값을 정의한다.
- ④ [Step 3], 테스트 Suite에 추가하고 다음 테스트를 실행하거나 신규로 생성한다.

The screenshot shows the 'Quest Code Tester: Builder' window. On the left, the 'Program Details' section shows 'Program: XML_USER.PK_XML_EXAMPLE' and 'Type: PACKAGE'. Below this is a table of subprograms and their test case status:

Subprogram	Test Cases Existing / New
Function GET_CUSTOMERS	1 / 1
Function GET_PRODUCTS	1 / 1
Procedure GET_ORDER	2 / 2
Function GET_EXEPECTED_RESULT	0 / 0
Procedure CREATE_ORDER	0 / 0
Procedure ADD_ORDER_ITEM	0 / 0
Procedure CHANGE_ORDER_ITEM	0 / 0
Procedure SET_DELIVERY_ADDOR...	0 / 0

The main window is titled 'Test Case Editor for Get Order'. It has three tabs: 'Description', 'Declarations', 'Subprograms', 'Initialization', and 'Cleanup'. The 'Description' tab is active, showing three steps:

- Step 1. Enter test case properties.** (Circled 2) This step includes a 'Description' field with the text '<NEW TEST CASE DESCRIPTION>'.
- Step 2. Set inputs for the execution of your program.** (Circled 3) This step includes a table for defining input values:

Name	Data Type	Value
P_ORDER_ID	PLS_INTEGER	1

- Step 3. Define outcomes that verify behavior of your program.** (Circled 4) This step includes a table for defining expected results:

A. Data changed by program	3. Operator	C. Expected result
Argument or RETURN Clause: P_XML_DOC - XMLTY...	root equals	Varchar2(32767) Order
Argument or RETURN Clause: P_XML_DOC - XMLTYPE	xpath scalar equals	Varchar2(32767) 1