Top Five Reasons to Choose Toad® Over SQL Developer

Written By:

John Pocknell
Senior Product Manager
Quest Software
# Contents

Abstract ............................................................................................................................................ 2

Introduction ...................................................................................................................................... 3

Toad for Oracle ................................................................................................................................. 5

SQL Developer ................................................................................................................................. 7

Top Five Reasons to Choose Toad Over SQL Developer ................................................................. 8

Reason One: Quality Matters ......................................................................................................... 8

  Expert Advice for Code Correctness, Readability, and Maintainability ....................................... 8

  Understanding and Documenting Code ........................................................................................ 10

  Functional (Unit) Testing .............................................................................................................. 11

Reason Two: Performance Matters ................................................................................................. 12

  Alternative Execution Plans ......................................................................................................... 13

  Ease of Use .................................................................................................................................. 15

  Identifying Problematic Statements ............................................................................................ 15

Reason Three: Maintainability Matters .......................................................................................... 16

  Understanding the Impact of Proposed Code Changes ............................................................... 16

  Ensuring Standards Are Followed ............................................................................................... 17

Reason Four: Team Collaboration Matters .................................................................................... 19

Reason Five: Productivity Matters ................................................................................................. 20

  Seamless Functionality .............................................................................................................. 20

  Customizable Interface ............................................................................................................... 21

  Automation .................................................................................................................................. 22

Conclusion ....................................................................................................................................... 24

For More Information ..................................................................................................................... 25

About the Author ............................................................................................................................. 26
Abstract

SQL Developer is free, so why should you choose Toad®? This technical brief reveals the top five reasons.
Introduction

Toad was one of the first Oracle integrated development environments (IDE) of its kind. Written in 1995 by a developer for developers, Toad® for Oracle is now used by more than two million people worldwide, including developers, DBAs, analysts, managers, support engineers, and many others. Toad has a cult following of extremely loyal users. Their enthusiasm is apparent at events like Oracle Open World and in countless Toad user groups around the world. And while many software companies have tried to emulate Toad’s success, it’s fair to say that all have fallen short. Toad remains the benchmark by which other tools are measured, including Oracle’s SQL Developer.

SQL Developer was introduced through an early adopter release in December of 2005. It attracted huge interest, not just because it’s free, but also because it was the first Oracle PL/SQL development IDE from the database vendor – even though JDeveloper already contained a lot of PL/SQL development features. SQL Developer has a dedicated user community, but even half a decade after the product’s inception, the list of enhancement requests – for features already provided in Toad for Oracle – continues to grow.

This technical brief answers the question, “If SQL Developer is free, why should I still choose Toad?”

The following table summarizes the main differences between Toad for Oracle and SQL Developer based on five core areas of Oracle database development:
<table>
<thead>
<tr>
<th></th>
<th>Toad for Oracle v11.0 (various editions)</th>
<th>SQL Developer 3.0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Code Quality</strong></td>
<td>Toad offers automated rules-based code reviews, through its Code Analysis feature, which can have coding violations displayed dynamically in the Editor. Coding rules can be adapted to suit different projects. All rules have been approved by Quest’s own Steven Feuerstein, one of the world’s leading experts on PL/SQL best practices. Code reviews can be scheduled and a dashboard provides managers with a project-wide view. Toad also provides the ability to build test cases automatically and store them.</td>
<td>Other than basic code templates and formatting, which Toad already offers, it is our understanding that SQL Developer offers little that specifically addresses code quality and best practices. Absence of code quality assessments will lead to an increase in unplanned development cycles. SQL Developer provides integrated functional testing, but test cases must be built manually before they can be used, which is labor-intensive.</td>
</tr>
<tr>
<td><strong>Application Performance</strong></td>
<td>Toad enables developers to identify and tune PL/SQL and SQL statements using a very simple interface or a comprehensive optimization environment. This ensures that development can take care of application performance issues rather than simply relying on DBAs.</td>
<td>SQL Developer appears to offer SQL tuning based on Oracle Tuning Advisor. This technology provides advice for the developer but may often fail to provide the most optimal solution. Tuning Advisor requires OEM Diagnostics and Tuning Packs, part of Enterprise Edition and must be licensed.</td>
</tr>
<tr>
<td><strong>Code Maintainability</strong></td>
<td>Code that is not maintainable will often lead to delays when it has to be modified. Coding best practices, as recommended by our experts like Steven Feuerstein, and implemented in the Code Analysis feature, ensure coding changes are simplified, improve consistency, and reduce errors.</td>
<td>Other than basic code templates and formatting, which Toad already offers, it is our understanding that SQL Developer offers little that specifically addresses code maintenance. This is likely to lead to an increase in errors and problems in production.</td>
</tr>
<tr>
<td><strong>Team Collaboration</strong></td>
<td>Teams are more effective units than separate individuals. Toad makes it easy to collaborate settings, database connections, templates, and code-review standards among team members; this improves consistency, minimizes errors, and maximizes productivity.</td>
<td>Other than the possibility to share some templates, it appears that SQL Developer is really designed just for the individual developer; it offers little opportunity to collaborate effectively with other developers in a team.</td>
</tr>
<tr>
<td><strong>Productivity</strong></td>
<td>With its functional depth, superior workflow, and task automation, time savings in Toad can be measured in hours, not minutes. Many tasks – such as data exports, reports, schema comparisons, code reviews, etc. – can be performed over again minimizing errors; entire multi-task workflows can be built and scheduled.</td>
<td>SQL Developer’s workflow seems reasonably logical, but, in terms of functional depth, developers may quickly find that it offers little help for performing some essential tasks. So far as we understand, SQL Developer does not offer any task automation capabilities.</td>
</tr>
</tbody>
</table>
Toad for Oracle

From its early days as an Oracle developer’s tool, the Toad for Oracle offering has expanded into new editions that are built to suit different job functions, as shown in Figure 1. Most offerings are available on several database platforms in addition to Oracle, including SQL Server, IBM DB2, and MySQL. Quest offers configurations and licensing models to support individual users and large corporations alike.

![Figure 1. Toad comes in different editions to support a variety of job functions.](image)

Toad also supports newer technologies such as cloud databases, NoSQL, and Hadoop. Toad® Extension for Visual Studio, which supports Microsoft’s Visual Studio 2010 application lifecycle methodology, Toad® Extension for Eclipse, and Toad® for Cloud Databases have all been released, and more Toad editions are in the works.
Figure 2. Toad has extended its reach to new technologies such as cloud databases, NoSQL, and Hadoop.
SQL Developer

Launched in 2005 and based on the same platform as JDeveloper, SQL Developer is targeted to the individual Oracle PL/SQL developer. It provides a reasonable level of functionality to accomplish development tasks such as building, testing, debugging, and Oracle 11g hierarchical profiling.

Figure 3. SQL Developer is designed for the individual developer.

SQL Developer also provides read-only access to database platforms other than Oracle and interfaces with Oracle's Times Ten in-memory database and Oracle's Application Express (APEX).
Top Five Reasons to Choose Toad Over SQL Developer

Make no mistake: Oracle SQL Developer is not the same as Toad for Oracle. Often, people compare both tools and draw that incorrect conclusion without investigating the real differences.

Let’s face it, all development tools should allow users to browse objects in the database, view and change data in an Oracle table, as well as provide an editor for building code, a debugger, a code profiler, a reporting tool, the means to export and import data, and a graphical query builder. Since the introduction of Toad, these things are now expected from any development IDE. But these features alone should not be used as the basis for an effective comparison between one tool and another, especially when the integrity and performance of critical business applications is at stake.

This paper examines all editions of Toad for Oracle v11.0 and provides a representative comparison with Oracle SQL Developer v3.0 from a productivity standpoint. This paper is not about the features that SQL Developer lacks; it goes beyond that to analyze the functionalities that matter most to our users.

And how do we know these capabilities matter? Well it’s a fact that Oracle has one of the largest implementations of Toad worldwide. We saw Toad customers move to SQL Developer when it was introduced, because it’s free and was thought to offer the same functionality as Toad. But many of those customers have now returned to Toad after realizing they cannot accomplish in SQL Developer what they can in Toad. That alone says a lot about the value Toad offers our customers – value they need to work more effectively.

So why should you choose Toad? Because unlike other tools, Toad improves what really matters when it comes to application code: quality, performance, maintainability, team collaboration, and productivity.

Reason One: Quality Matters

Expert Advice for Code Correctness, Readability, and Maintainability

One of the characteristics that makes Toad (and indeed Quest Software) unique is our wealth of domain expertise. Our experts are fluent in Oracle, SQL Server, DB2, and other platforms. In the case of Toad for Oracle, those experts include Steven Feuerstein, one of the world’s leading authorities in PL/SQL. Many of Steven’s coding best practices are embodied in Toad features, ranging from helpful knowledge and advice to flexible and automated code reviews.

At any point during code creation or change, a Toad developer can receive expert feedback. Code Analysis, Toad’s PL/SQL and SQL advisor, will analyze the code, present coding violations dynamically in the Editor, as well as offer detailed corrective recommendations on how to optimize it – not only for code correctness, but also for better readability and maintainability, as well as other best practices categories. This extends Toad’s benefits to entire development and quality assurance teams. So when the developer moves on to something else, the person who inherits that same program is not left to wonder what is going on.
For example, if the database is not complaining with ORA or PLS errors when you create your stored procedures, it is very tempting to just assume that everything must be OK. With Toad, you can address that nagging fear in the back of your head that maybe all is not well with your code. With Steven’s PL/SQL best practices built into Toad’s parser, the developer has instant access to better programming techniques and theory.

![Image: Toad offers practical steps to improve your PL/SQL programs.](image)

Code Analysis lets the development team choose rules and alter rule characteristics based on their company or project requirements. The Edit Code Analysis Rules enable the rules characteristics to be modified, stored, and then saved to each team member.
We like to think of Toad as a coding coach that helps less experienced developers become experts with the advice of seasoned professionals like Steven. In contrast, SQL Developer offers nothing to encourage developers to write smarter code. This often results in code that is sub-standard and harder to maintain.

Understanding and Documenting Code

For developers who live in the maintenance cycle, Toad also has the ability to document what a program does: It will show the objects involved and the data touched by each execution of the program. Testing is a much more fruitful exercise when you know exactly what to test!
Functional (Unit) Testing

Another way that application code quality can be ensured is through functional (or unit) testing of PL/SQL programs. Both Toad and SQL Developer offer unit testing:

- The Quest® Code Tester for Oracle tool is a component of the Toad® Development Suite for Oracle. Written by Steven Feuerstein, it enables easy test case creation that can be automated based on the developer’s knowledge of expected behavior and outcomes.
- Oracle SQL Developer’s unit testing is available as part of the IDE, but its functionally is limited. In particular, it requires the user to figure out how to create the test cases, so it isn’t automated testing by any stretch of the imagination.

With Quest Code Tester, the user can click through test scenarios or use out-of-the-box tests. For example, you can test a scenario in which row deletion from a table fails due to a primary key relationship violation.

Figure 7. Test your code just like the professionals do with Feuerstein’s built-in functional tests.
A good testing tool is invaluable because it encourages more testing. Developers typically dislike testing and avoid it; the first roadblock they encounter will only steer them back to activities they do enjoy, such as writing more programs. Toad can easily test random or specific inputs across multiple datatypes and provide 26 ways to test the state of a table. The following figure shows how easy it is to set up date inputs:

**Input Value for IN YEAR**

The DATE datatype allows you to manipulate date-time information with time precision to the nearest second. As an input or outcome, you can provide a literal value or specify an expression. You can also choose a value from the pre-defined Test Data Groups.

**Argument Representation in Test Code**

Variable used in generated code for this value: \_IN\_YEAR

Datatype of the variable: DATE

**Specify a list of values or a query that returns a set of values, which will be used to generate multiple test cases.**

- Use existing list or query.
- Create a new list or query.

**Description:**

The set of date values that you would typically use to test a variety of boundary conditions, such as the beginning and ending of months and years.

**Specify Values Below:**

<table>
<thead>
<tr>
<th>Position</th>
<th>Name</th>
<th>Value</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NULL Date</td>
<td>TO_DATE (NULL)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Current System Date/Time</td>
<td>CURRENT_DATE</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Current System Date -</td>
<td>TRUNC (CURRENT_DATE)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>First day of this year</td>
<td>TRUNC (TO_DATE ('01-01'))</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Last day of this year</td>
<td>TRUNC (TO_DATE ('12-31'))</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>First day of this month</td>
<td>TRUNC (TO_DATE ('01-01'))</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Last day of this month</td>
<td>LAST_DAY (TO_DATE ('01-31'))</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Same date, previous year</td>
<td>ADD_MONTHS (CURRENT_DATE)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Same date, next century</td>
<td>ADD_MONTHS (CURRENT_DATE)</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 8. Need help setting up date inputs? It simply does not get easier than this.**

The latest version of Code Tester includes a feature called Run-To-Test, which enables developers to run an existing program and save the input arguments and other information as a new stored test case.

Toad’s intuitive testing features ensure that developers will be more successful at building tests, which will directly result in better programs that can be developed and maintained at a lower cost.

**Reason Two: Performance Matters**

Application performance is another Toad strength recognized by leading industry analysts. Development teams that fail to address code performance issues early on face potentially serious and expensive problems in production.
Alternative Execution Plans

It takes many years of experience to identify and resolve SQL execution performance problems in scripts and PL/SQL code – unless you have the right tools. Toad has you covered. It enables you to see execution plans for code that’s being developed, explore code that’s already executing, or find poor plans from the SGA.

Many tools, including SQL Developer, make execution plans for SQL statements readily available, but only Toad has the ability to suggest rewrite alternatives. Start with one problematic SQL statement, let Toad have a swing at it, and you now have several dozen SQL queries and plans to choose from. While the Oracle Enterprise Manager optimizer is quite powerful and gets smarter with each release, it is still far from being able to take any request for data and build the ideal execution plan. Cue the Toad component, Quest® SQL Optimizer for Oracle.

Figure 9 shows how, in seconds, Toad was able to identify multiple rewrite alternatives with unique plans available from a single SQL statement.

![Figure 9. Toad quickly provides alternate rewrites for a problematic SQL statement.](image)
Seeing each re-write scenario side by side with the original query makes the evaluation process a cinch.

Figure 10. Toad displays each alternative side by side with the original query to make comparison easy.
Ease of Use

Want a really simple interface for all this technology? We’ve got that covered, too. Simply activate the Auto Optimize SQL feature in the Toad Editor by clicking the Auto Optimize SQL button to enjoy the exact same optimization capabilities – without having to open the Quest SQL Optimizer tool separately.

Identifying Problematic Statements

Of course it’s not enough to be able to just automatically tune a statement; developers need assistance with identifying which statements need to be tuned. Rather than tune every statement they write, Toad can categorize plans as problematic or complex and ripe for tuning. Development managers or QA engineers can also use Toad’s Scan SQL feature to pull out all the problematic plans from an application project or from the database itself. We have a wealth of experience in the field of application performance tuning and this expertise is leveraged in Toad to suit varying levels of expertise.

Although SQL Developer 3.0 now offers the capability to identify and tune SQL statements in scripts or PL/SQL code, this actually leverages their OEM Tuning and Diagnostics Packs, which are currently available through OEM Database Control and require separate licensing. Contrary to what you may have heard, Toad for Oracle does not require licensing of OEM Diagnostics and Tuning Packs to utilize its SQL tuning and optimization functionality.
Reason Three:  Maintainability Matters

Understanding the Impact of Proposed Code Changes

As Steven often reminds us, the code you write today will have to be maintained for years into the future. What he is saying is that, although developers often just want to write code as quickly as possible, this is actually a short-sighted approach. Code that cannot be effectively maintained will ultimately cost the business in the future. For instance, if developers who maintain code do not understand how a change to a stored procedure or package can potentially affect other database objects, then a simple change could be disastrous to the underlying application.

We cannot rely on developers to produce maintainable code, even if they are provided with knowledge and advice. The tool must seamlessly facilitate the creation of maintainable code, or it won’t happen. Toad’s Code Road Map feature provides developers with a graphical representation of their code and the objects to which it relates. This ensures that developers understand the potential impact of changes to code (or to objects like tables) that could break the application.

Figure 12. Code Road Map ensures that developers understand the bigger picture.

Oracle SQL Developer provides no such assurances; developers must figure out whether a proposed code change will impact something else. With so much code to maintain across your applications, are you prepared to take that chance? Of course not; it’s like driving without car insurance.
Ensuring Standards Are Followed

Each organization must decide what maintainability rules are appropriate for them, but all organizations need to ensure that their rules are implemented consistently. SQL Developer only offers code templates to improve consistency and some basic code refactoring. These methods alone do not ensure that developers create code that is easy to maintain.

Toad offers far more. In particular, development managers and team-leads can easily perform a comprehensive code quality review of all the PL/SQL and SQL in the application. This enables the manager to measure the quality of the code from the beginning and track it throughout the development lifecycle. Poorly implemented code can be highlighted and returned to the developer with advice on how to make improvements. HTML reports can be published via email. Code reviews can be automated and scheduled using Toad’s automation features, saving hours of time, increasing efficiency, and ensuring consistency.

The ability to maintain quality from the beginning of a software project is even more critical when you are working with contractors or remote development teams. Toad shines a bright light on application code and makes it harder for inefficient code to sneak into the database.

Figure 13. Toad provides every development manager’s dream: a consolidated code review dashboard.
### EXAMPLE_PROC_INDEXING

#### Results Summary (All Rules by Objective)

<table>
<thead>
<tr>
<th>General</th>
<th>Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Object: EXAMPLE_PROC_INDEXING</td>
<td>Passed: 91</td>
</tr>
<tr>
<td>Run Date: 09/15/2011 at 10:20:03 AM</td>
<td>Flagged: 5</td>
</tr>
<tr>
<td>Rules Included: 96 of 134</td>
<td>Errors: 0</td>
</tr>
<tr>
<td>Total Rule Violations: 7</td>
<td></td>
</tr>
</tbody>
</table>

#### Code Analysis Violations

The table below displays Code Analysis Violations for the selected item.

<table>
<thead>
<tr>
<th>Rule ID</th>
<th>Rule Definition</th>
<th>Rule Category</th>
<th>Rule Objective</th>
<th>Rule Security</th>
<th>Line Number</th>
<th>Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>5602</td>
<td>Avoid coding procedures with no parameters.</td>
<td>Procedure</td>
<td>Program Structure</td>
<td>Information</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>2610</td>
<td>Limit use of weak REF CURSOR types.</td>
<td>Cursor</td>
<td>Maintainability</td>
<td>Information</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4602</td>
<td>Avoid use of literals in non-declarative parts of the program.</td>
<td>Literal</td>
<td>Maintainability</td>
<td>Information</td>
<td>9</td>
<td>32</td>
</tr>
<tr>
<td>4602</td>
<td>Avoid use of literals in non-declarative parts of the program.</td>
<td>Literal</td>
<td>Maintainability</td>
<td>Information</td>
<td>10</td>
<td>30</td>
</tr>
</tbody>
</table>

**Figure 14.** Toad provides a management summary report that highlights issues.
Reason Four: Team Collaboration Matters

Good teamwork can improve application quality, reduce program bottlenecks, and minimize maintainability issues. But in many development shops, teamwork is still a challenge. Many times, when we ask customers how effectively coding standards are used by their development teams, they smile and say, “That’s a challenge for us.”

Toad offers a variety of collaboration tools. For example, the Project Manager utility (shown below) allows you to organize databases, connections, database objects, scripts, data models, to-do items, favorite SQL commands, FTP, and local folders and files—all in a single interface. This makes it extremely easy for a team lead to bootstrap new developers with everything they need to hit the ground running.

Toad also provides everyone on your team with instant access to frequently used database objects, so they can perform routine tasks in a single click instead of navigating the vast database looking for what they need.

Oracle SQL Developer, as mentioned earlier, was designed as a development tool for the single developer; it provides minimal abilities to share things with other developers; these are limited to code snippets, code templates, and formatter templates.
Reason Five: Productivity Matters

Seamless Functionality

Toad is a productivity tool at heart. Quest recognizes that database professionals spend as much time outside the database as they do inside, so when you need to back up your code or objects, Toad allows you to simply drag and drop an object to an FTP folder.

For editors, the essential Toad window is the Editor, where they build and edit code. Productivity features in the Editor aren’t limited to keyboard shortcuts; the Editor provides the time-saving features and functions of Toad in such a seamless manner that you don’t even have to think about it. When customers are asked what they like about Toad, they often reply, “Toad helps me get on with what I need to do without getting in my way.” This could easily be the mantra of Toad’s development team.

For example, SQL Developer’s Worksheet and Toad Editor both provide contextual lookups for developers as they type. Toad for Oracle even provides contextual lookups on parameter and variable names in PL/SQL programs as well as lookups for PL/SQL package calls containing multiple arguments.

Figure 16. Toad for Oracle provides auto lookup of parameter and variable names.

Toad for Oracle also enables developers to quickly extract embedded SQL statements from their PL/SQL for execution (<Shift><F9>) or tuning without having to manually reform it, saving time and reducing errors.
Figure 17. Toad automatically extracts and executes embedded SQL from PL/SQL code

Customizable Interface

Because Toad is used by so many people in a variety of roles, we provide a simple way to customize the user interface to suit your needs. Items can be quickly removed from, moved within, or added to the standard toolbar and menu to suit your individual style and maximize productivity. You can configure other Toad features as well. For instance, if you want help on specific database object and column names while developing code, but only on demand, you can configure that behavior. If you are new to Oracle and/or SQL, the Editor can be configured to suggest keywords and expected tokens for commands that are written; experts can disable this feature.

Figure 18. Toad’s Editor can be configured to suggest keywords and expected tokens for commands that are written; experts can disable this feature.
Automation

Toad can automate tedious tasks, such as writing the same SQL over and over again. And it can also make complicated tasks as simple as one click. Toad’s Rerun and Automation Designer features make any combination of the following tasks repeatable, either on demand or scheduled via Windows:

![Automation Example](image)

Figure 19. Many routine tasks can be stored and re-used, saving hours.

The following are some of the tasks you can automate with Toad:

- Executing one or more SQL statements
- Reviewing code using Code Analysis
- Saving the results of queries or contents of tables to Excel
- Importing data from a file to the database
Top Five Reasons to Choose Toad® Over SQL Developer

- Comparing and synching databases and schema-level objects
- Performing database health checks
- Generating various reports
- Integrating email and FTP delivery of results and reports
- Generating DDL scripts for a single object or an entire schema

Automating these kinds of tasks marks the difference between a good developer and an expert developer. SQL Developer, which delivers none of this automation, might be an adequate tool for some, but if you are too busy to do all the work manually, then Toad can be invaluable.
Conclusion

Just because a tool is free, that doesn’t mean it will save you money. In fact, it could lull you into a false sense of security. If you’re solely focused on productivity gains and cost savings, you may overlook other essential features like quality, performance, and maintainability – and actually spend more money as a result. While SQL Developer struggles with delivering basic, must-have features, Toad goes beyond the necessities, delivering a wealth of invaluable features, plus many others that you never imagined were possible.

Oracle itself has one of the largest implementations of Toad worldwide. Quest saw a number of Toad customers move to SQL Developer when it was introduced because it’s free and they hoped it would offer the same functionality as Toad; however, many of those customers have now returned to Toad because they cannot accomplish in SQL Developer what they can in Toad. This is proof positive that customers – and even Oracle – can’t work effectively without Toad.
For More Information

For an interesting article on the hidden costs of freeware tools, see this white paper by John Weathington, President and CEO of Excellent Management Systems, Inc.

For more information on how Toad for Oracle compares with other IDEs, read this article by Jeff Smith.

John Weathington offers an independent assessment of Toad for Oracle v10.0.

You can also read numerous case studies from customers on the value of using Toad for Oracle.

Another excellent source of information on Toad for Oracle (and all the products associated with Toad) is Toad World. Toad World is your one-stop shop for all things Toad!

For an update on the latest release of Toad for Oracle, version 11, please read the new technical brief.
About the Author

John Pocknell is a senior product manager at Quest Software. Based at the European headquarters in the U.K., John is responsible for Toad for Oracle and other products in the Toad family. He has been with Quest since 2000, working in the database design, development, and deployment product areas, and he has run many Toad training courses for customers. John has spent the last 11 years successfully evangelizing Toad to customers at various events throughout Europe, the U.S., and the Asia-Pacific region. John writes many blogs and papers that are published on the Toad user community, Toad World.

John has worked in IT for more than 25 years, most of that time in Oracle application design and development. He is a qualified aeronautical engineer with more than 10 years provisioning IT consultancy services and implementing quality assurance systems to ISO 9001.
Top Five Reasons to Choose Toad® Over SQL Developer

© 2012 Quest Software, Inc.
ALL RIGHTS RESERVED.

This document contains proprietary information protected by copyright. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose without the written permission of Quest Software, Inc. (“Quest”).

The information in this document is provided in connection with Quest products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest products. EXCEPT AS SET FORTH IN QUEST’S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software World Headquarters
LEGAL Dept
5 Polaris Way
Aliso Viejo, CA 92656
www.quest.com
email: legal@quest.com

Refer to our Web site for regional and international office information.

Trademarks

Quest, Quest Software, the Quest Software logo, AccessManager, ActiveRoles, Aelita, Akonix, AppAssure, Benchmark Factory, Big Brother, BridgeAccess, BridgeAutoEscalate, BridgeSearch, BridgeTrak, BusinessInsight, ChangeAuditor, ChangeManager, Defender, DeployDirector, Desktop Authority, DirectoryAnalyzer, DirectoryTroubleshooter, DS Analyzer, DS Expert, Foglight, GPOADmin, Help Desk Authority, Imceda, IntelliProfile, InTrust, Invirtus, iToken, iWatch, JClass, Jint, JProbe, LeccoTech, LiteSpeed, LiveReorg, LogADmin, MessageStats, Monosphere, MultiSess, NBSpool, NetBase, NetControl, Npulse, NetPro, PassGo, PerformaSure, Point, Click, Done!, PowerGUI, Quest Central, Quest vToolkit, Quest vWorkspace, ReportADmin, RestoreADmin, ScriptLogic, Security Lifecycle Map, SelfServiceADmin, SharePlex, Sitraka, SmartAlarm, Spotlight, SQL Navigator, SQL Watch, SQLab, Stat, StealthCollect, Storage Horizon, Tag and Follow, Toad, T.O.A.D., Toad World, vAutomator, vControl, vConverter, vFoglight, vOptimizer, vRanger, Vintela, Virtual DBA, VizionCore, Vizioncore vAutomation Suite, Vizioncore vBackup, Vizioncore vEssentials, Vizioncore vMigrator, Vizioncore vReplicator, WebDefender, Webthority, Xaffire, and XRT are trademarks and registered trademarks of Quest Software, Inc in the United States of America and other countries. Other trademarks and registered trademarks used in this guide are property of their respective owners.

Updated January 2012
About Quest Software, Inc.

Quest Software (Nasdaq: QSFT) simplifies and reduces the cost of managing IT for more than 100,000 customers worldwide. Our innovative solutions make solving the toughest IT management problems easier, enabling customers to save time and money across physical, virtual and cloud environments. For more information about Quest solutions for administration and automation, data protection, development and optimization, identity and access management, migration and consolidation, and performance monitoring, go to www.quest.com.

Contacting Quest Software

PHONE 800.306.9329 (United States and Canada)
If you are located outside North America, you can find your local office information on our Web site.

EMAIL sales@quest.com

MAIL Quest Software, Inc.
World Headquarters
5 Polaris Way
Aliso Viejo, CA 92656
USA

Contacting Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a commercial version and have a valid maintenance contract.


SupportLink gives users of Quest Software products the ability to:

- Search Quest’s online Knowledgebase
- Download the latest releases, documentation and patches for Quest products
- Log support cases
- Manage existing support cases

View the Global Support Guide for a detailed explanation of support programs, online services, contact information and policies and procedures.

TBD-ToadOverSQLDev-US-SW-01162012