

Speed Your DevOps Pipeline by Including Database Development

Toad® DevOps Toolkit removes the database development speed bump from your Oracle DevOps process by enabling automated PL/SQL testing and static code reviews during application deployment.

By John Pocknell, senior product manager, Quest Software



ABSTRACT

To succeed today, organizations need to be able to respond quickly to emerging market demands. That means being flexible, collaborating across departments and making use of the latest technologies. Accordingly, many companies are implementing innovative DevOps processes to automate key aspects of the application development lifecycle and deliver new releases faster and with fewer defects. Unfortunately, database development is often excluded from these initiatives because the time-consuming code testing and reviews it requires would put a huge speed bump in the DevOps process.

This tech brief offers a solution to this conundrum: Quest® Toad® DevOps Toolkit. This Windows-based application enables you to bring Oracle database development into your DevOps pipeline, making for even better and more reliable application deployment.

INTRODUCTION

Organizations today are under tremendous pressure to build, test and release new software and updates as fast as they can, while ensuring their code is completely bug-free. DevOps processes — agile methodologies based on continuous integration (CI) and continuous deployment (CD) — can help keep things rolling along while ensuring quality.

DevOps, a mashup of “development” and “operations,” is best understood as a business initiative: to improve communication and collaboration among stakeholders in order to increase the speed and quality of software deployment. To achieve these goals, DevOps requires cultural change, automated processes and investment in the right technologies

From a software engineering perspective, this involves automation and strict monitoring throughout the entire development lifecycle, from integration, testing and release to deployment and infrastructure management. The result is shorter development cycles, higher-quality updates and more dependable releases that better align with current business strategies (see Figure 1).

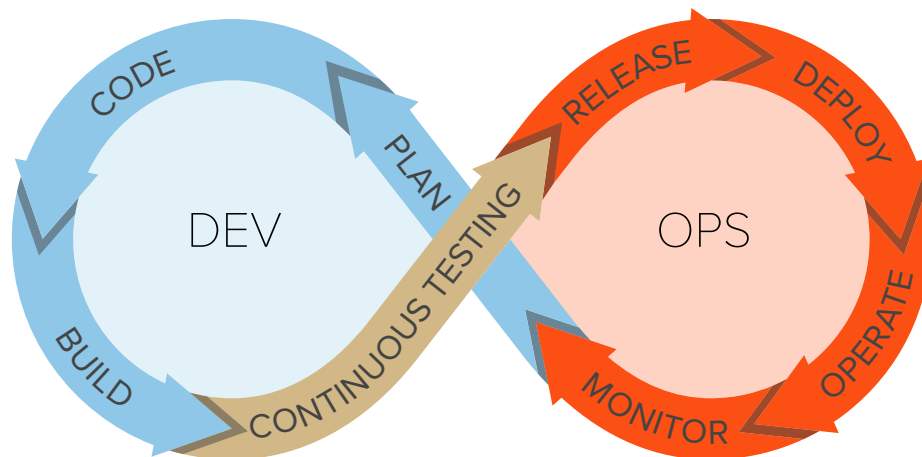


Figure 1. DevOps applies automation and strict monitoring throughout the development lifecycle to enable shorter development cycles and higher quality releases.

With Toad DevOps Toolkit, you can execute key database development functions within your DevOps workflow — without compromising quality, performance or reliability.

Beyond helping organizations meet the increasing market demand for “faster, better, more,” DevOps fosters a culture of collaboration between teams — the software developers and the IT infrastructure folks — that historically functioned in siloes. It’s been a win all around; organizations achieve increased trust, faster software releases, better problem-solving and more business agility.

But there is one elephant in the room: How can the database side of things be integrated into this sleek, streamlined development process? Because many critical business applications are backed up by large databases, excluding that piece of the puzzle is like renovating a kitchen but shutting off the water supply.

If you’re a DevOps veteran, you are probably laughing right now. Database development has not traditionally been included in the DevOps process because of both its complexity and the sensitive nature of all that data living in the databases. Many organizations simply don’t want to move too quickly when critical business data could be put at risk. Also, database development requires a tremendous amount of very time-consuming unit testing and code analysis. When organizations have tried to include database development, it’s been a huge bottleneck in their otherwise agile DevOps workflow.

TOAD DEVOPS TOOLKIT

Enter Quest Toad DevOps Toolkit. This easy-to-implement solution addresses the challenges of bringing Oracle database operations into the DevOps infrastructure by making it possible to execute key database development functions within your DevOps workflow — without compromising quality, performance or reliability.

This unique Windows-based application exposes key Toad for Oracle functionality as programmable objects. As a result, DevOps automation tools, such as Jenkins and Bamboo, can include database-related steps as part of continuous integration processes via PowerShell or Visual Basic scripts — speeding project delivery while ensuring code quality (see Figure 2).

Specifically, Toad DevOps Toolkit makes it possible to integrate the following database development tasks into the DevOps framework:

- **PL/SQL code unit testing** — To ensure functional correctness and ensure applications do not break in production
- **Code reviews** — To make sure code meets standards for quality and maintainability
- **Database compare, schema compare, and data compare and sync** — To ensure downstream database environments match the source database to reduce errors and risk of failure

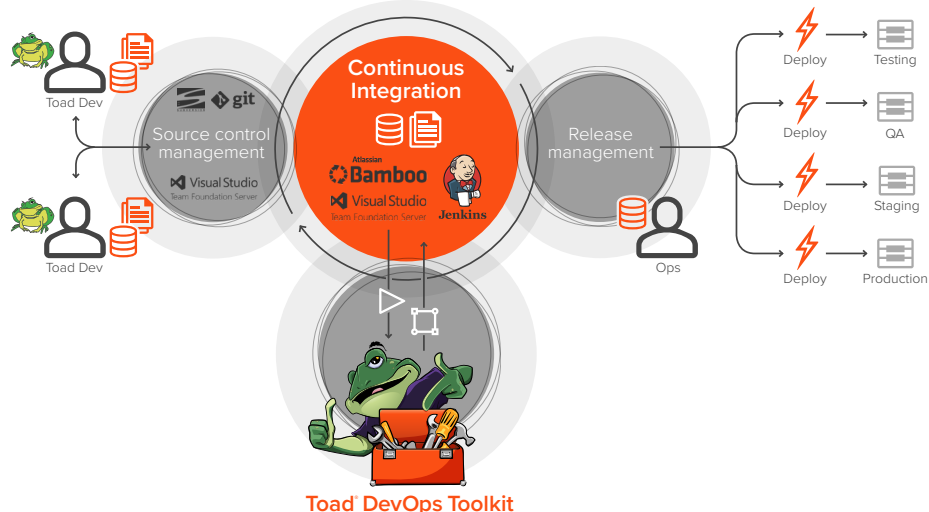


Figure 2. Bring database development into your DevOps pipeline with Toad DevOps Toolkit.

- **Script execution** — To physically deploy database changes throughout the release pipeline

And it's incredibly flexible: Toad DevOps Toolkit can be installed on any Windows-based laptop, desktop, server or VM, either on premises or in the cloud, provided that device can connect to the build automation server.

As function calls are made into Toad DevOps Toolkit to perform the tasks listed above on the database objects in the build, output, such as reports, JSON

content and logs, can be specified and viewed within the build automation console, where all operations are orchestrated (see Figure 3).

KEY FEATURES

Faster DevOps pipeline

Increasing the velocity of the DevOps pipeline creates greater business agility in meeting rapidly changing market demands; however, this agility can be achieved only if database teams and application teams are aligned and share the same processes and systems. With

Toad DevOps Toolkit can be installed on any Windows-based laptop, desktop, server or VM, either on premises or in the cloud, provided that device can connect to the build automation server.

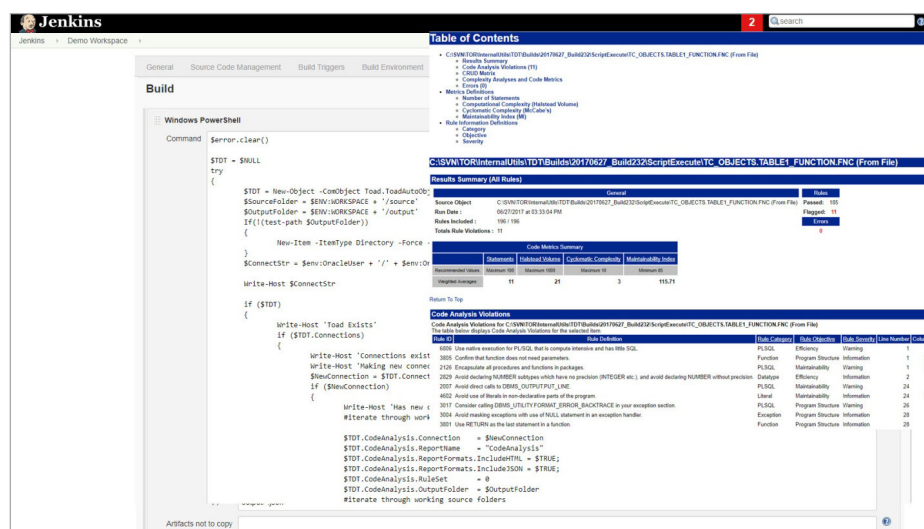


Figure 3. A static code review report generated by a Toad DevOps Toolkit Code Analysis call, being viewed in a Jenkins build console

Toad DevOps Toolkit exposes key Toad for Oracle functionality as programmable objects so you can automate processes and apply them consistently.

Toad DevOps Toolkit, you can synchronize database and application changes within your DevOps processes to more easily meet deadlines across the deployment lifecycle.

Better code quality

A successful DevOps program must reduce code defects in production. Achieving this goal requires a proactive approach to quality, which can be accomplished only by using repeatable, objective standards. Toad DevOps Toolkit exposes key Toad for Oracle functionality as programmable objects so you can automate processes and apply them consistently. Built-in pass/fail notifications indicate whether a code build is of sufficient quality to be deployed to production.

PL/SQL unit testing

It has always been important for developers to test their code for correct function under use cases that the code is expected to encounter when it's called by the application. This unit testing becomes critically important in the context of an automated DevOps pipeline, where things are moving much faster. If database developers are brought into the DevOps process, the requirement for careful unit testing applies equally to their code.

But most database developers aren't going to suddenly start creating unit tests, especially for Oracle PL/SQL code, which can be technically challenging, labor-intensive and time-consuming. Unless unit testing is built in to the process and automated, it will be neglected, and eventually a piece of untested code will cause an application to break.

Fortunately, Toad for Oracle Developer Edition includes a dedicated PL/SQL testing solution — Code Tester for Oracle — that makes creating unit tests really simple, and you can even create unit tests by executing your code in the Toad Editor. And Toad DevOps Toolkit enables you to work with all the test suites, unit tests and test cases in your Code Tester for Oracle repository. You can run these tests as part of your continuous integration builds, provided a Code Tester repository is installed

on the system where Toad DevOps Toolkit is being used. You simply pass in the tests or test suites to be run and a list of objects to be tested; those tests are executed for the associated objects; and the results are passed back to you.

Here's how to integrate unit testing into your build automation process using Toad DevOps Toolkit:

1. Your team members create or make changes to their PL/SQL code using Toad for Oracle.
2. In Toad for Oracle or Code Tester for Oracle, they write their unit tests for those objects, and then check both the PL/SQL code and the associated tests into their version control system (VCS).
3. Your automated build process (such as Jenkins or Bamboo) checks for revised code in the VCS at configured intervals, and upon finding it, checks the code and unit tests out to a staging folder.
4. Within the build automation process, you prepare a script (using PowerShell or Visual Basic) that invokes Toad DevOps Toolkit, runs the tests, and outputs the results within the build automation console, indicating test pass or failure.

As in application regression testing, test failures need to be reported within the build process with a feedback loop to development, so they can address the issues before the build is deployed.

Static code quality reviews

Static code reviews are typically subjective, frequently lack depth and are often incomplete. Including these reviews as an effective part of an automated build process requires a rules-based approach that can be applied consistently across different code. Toad Code Analysis is an automated code review and analysis tool that enables individual developers, team leads and managers to ensure that the quality, performance, maintainability and reliability of their code meets and exceeds their best practice standards.

Toad DevOps Toolkit enables you to perform static code reviews based on predefined Toad Code Analysis rules. Pass/fail status notifications based on predefined thresholds will indicate whether the code is ready to move forward in the application development cycle.

```

1 # Start TDT
2 Write-Host 'Starting TDT'
3 $TDT = New-Object -ComObject 'Toad.ToadAutoObject'
4
5 # Set DB Login info
6 Write-Host 'Setting DB Login Info'
7 $SourceConnection = "quest_prod/prodXE"
8
9 # Set CA parameters
10 Write-Host 'Setting CA Parameters'
11 $TDT.CodeAnalysis.Connection = $TDT.Connections.NewConnection($SourceConnection)
12 $TDT.CodeAnalysis.ReportName = "CA Report"
13 $TDT.CodeAnalysis.OutputFolder = "C:\Temp"
14 $TDT.CodeAnalysis.ReportFormats.IncludeHTML = $TRUE
15
16
17 # Set DB Object Info to Analyze
18 Write-Host 'Setting DB Object Info to Analyze'
19 $DBObject = $TDT.CodeAnalysis.DBObjects.Add()
20 $DBObject.ObjectName = "OW_PLACE_ORDER"
21 $DBObject.ObjectOwner = "QUEST_PROD"
22 $DBObject.ObjectType = "FUNCTION"
23
24
25 # Execute Code Analysis
26 Write-Host 'Executing Code Analysis...'
27 $TDT.CodeAnalysis.Execute()
28
29 # Quit TDT
30 Write-Host 'Stopping TDT'
31 $TDT.Quit()
32

```

```

PS C:\Windows\system32> C:\JohnP\Training\SC Training\TOAD\Toad DevOps Toolkit\TDT_Simple.ps1
Starting TDT
Setting DB Login Info
Setting CA Parameters
Setting DB Object Info to Analyze
Executing Code Analysis...
Stopping TDT
PS C:\Windows\system32>

```

Figure 4. Calling the Code Analysis module in Toad DevOps Toolkit directly using a PowerShell script

Here's how to integrate Code Analysis into your build automation process using Toad DevOps Toolkit:

1. A team member (typically the development manager or team lead) uses Toad for Oracle to define the required code quality standards using a rule set (a collection of standard rules based on the knowledge of industry PL/SQL experts) within Code Analysis. This custom rule set file is then checked into the VCS. If you want to use one of the prebuilt rule sets that come with Toad DevOps Toolkit, you can skip this step.
2. To test your PowerShell scripts in advance, you can use Windows PowerShell ISE (a PowerShell utility) to invoke the Code Analysis module in Toad DevOps Toolkit, as illustrated in Figure 4.
3. Code Analysis performs an analysis on all of your changed code based on the rule set you specify in the script.
4. Based upon a predefined quality threshold, a pass/fail notification will be sent to the build automation console. If the code passes, a report is also generated (in HTML, XML or JSON format) and presented within the build automation console (see Figure 5).

Toad DevOps Toolkit enables you to perform static code reviews based on predefined Toad Code Analysis rules.

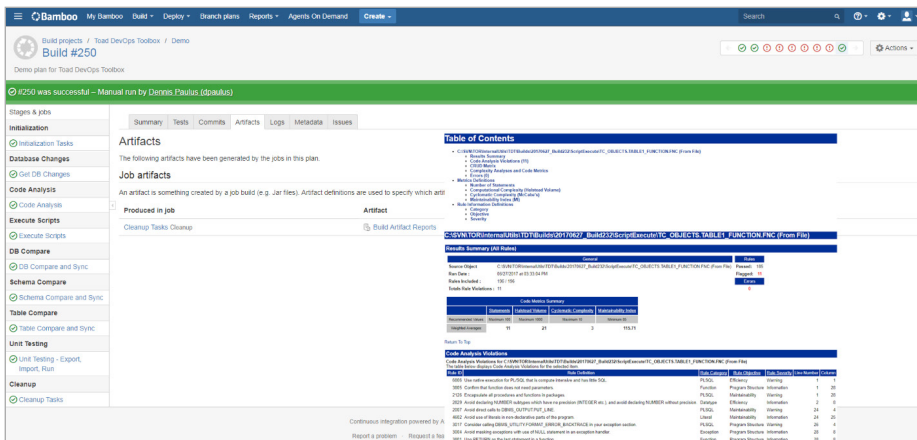


Figure 5. A static code review report generated by a Toad DevOps Toolkit Code Analysis call, viewed in a Bamboo build console

Difference Summary

As Of 6/27/2017 3:35:55 PM

Source: TC_OBJECTS@TDT_DEV

Target: TC_OBJECTS@TDT_TEST

Object Type	In Source Only	In Target Only	In Both with Differences
Package	0	1	0
Package Body	0	1	0
Table	0	0	2

Figure 6. Toad DevOps Toolkit can compare schema objects between two databases, produce a difference summary report like this one, and create a script to sync the objects.

Toad DevOps Toolkit can compare tablespaces, roles, users and schemas between two databases and create a synchronization script that updates the target database to match the source.

Exactly how do the Toad Code Analysis rule sets work? Code Analysis compares code against a set of best practices rules to establish consistent quality levels across all the code in a project. These rules are stored in rule sets. Toad DevOps Toolkit is installed with basic, built-in rule sets that you can use out of the box. However, you can adjust these rule sets to suit the requirements of different projects. You can also export rule sets from Toad for Oracle — or create and export custom rule sets — and then direct Toad DevOps Toolkit to use them.

Database compare and sync

Your CI system, through a prebuilt script that invokes Toad DevOps Toolkit, can compare database-level objects (such as tablespaces, roles and users) between two databases (such as dev and test) or snapshot files. You can configure Toad DevOps Toolkit to generate a synchronization script that updates the target database to match the reference source. You can run the script as part of the build or save it to a file.

Here's how to integrate a database compare into your build automation process using Toad DevOps Toolkit:

1. Developers make their schema changes and check them in to the version control system.
2. As a part of your build automation process, prepare a script (using PowerShell or Visual Basic) that invokes Toad DevOps Toolkit to compare the source database environment with the target database environment. The script will then generate a synchronization script, which checks

whether any changes were made to the target database outside of the CI process.

3. If the comparison shows a discrepancy, a difference report is generated and posted to the build automation console.
4. If the comparison shows the databases are identical, then schema changes within the build are potentially deployable to the target database.

This same process is repeated at each stage of the DevOps pipeline.

Schema compare and sync

Toad DevOps Toolkit can also compare schema objects between two databases, produce a difference summary report (as illustrated in Figure 6) and then create a sync script. The script can be run as part of a continuous deployment process or saved to a file that can be checked in to your version control system. Differences can be exported in one of several universal structured formats, such as JSON or XML, for further analysis as needed.

Data compare and sync

Toad DevOps Toolkit enables you to compare the data in individual tables or multiple tables and then generate a synchronization script for them. The script can be run as part of the CD process or saved to a file that can be checked in to your version control system.

Script execution

Following a successful database, schema and table compare between the source database and the target database, Toad DevOps Toolkit enables you to execute

any of the deployment scripts as part of your CD process. This enables you to implement changes by promoting build artifacts into your target environment and report success or failure status.

You can run individual scripts or specify a directory that contains multiple scripts to run. In addition, you can specify the output type, error options and the location of the script output.

Choice, ease and reliability

Toad DevOps Toolkit integrates with most continuous integration and continuous delivery tools, including Jenkins, Bamboo and Team Foundation Server, giving you unparalleled choice when it comes to integrating database development into your DevOps framework.

DevOps automation tools, such as Jenkins and Bamboo, can now include database-related steps as part of their continuous integration processes via scripts written in PowerShell or Visual Basic combining application and database deployments while ensuring code quality and reliability.

Because Toad DevOps Toolkit runs on Windows-based platforms, you'll have easy deployment with minimal disruption to your existing DevOps processes. And it doesn't hurt that this useful tool comes from such a trusted vendor. Toad DevOps Toolkit builds on the Quest team's long track record of delivering reliable, comprehensive and high-quality solutions, along with enterprise-level technical support.

TOAD DEVOPS TOOLKIT IN ACTION

Case study: financial services company

A financial services company based in Hartford, Connecticut, employs 9,000 developers in the United States and India. The business tasked the IT department with shortening its database release cycles from three months to just a few weeks. The CTO's team needed to figure out how to meet this goal for their Oracle development process without compromising quality, performance or scalability.

More than 1,000 developers in the organization are already using Toad for Oracle, but they have not been doing any PL/SQL testing or code quality reviews. As a result, they often need to rework aspects of their code when an application breaks, bogging down the development process. Many of the application teams are already using Jenkins with their Apache Subversion source control system, so it's just a matter of bringing everything together to create a more all-encompassing DevOps system.

The IT department is set to launch a pilot project using Toad Developer Edition with Toad DevOps Toolkit to streamline database development and bring it into the DevOps pipeline — which will help them meet their overarching business goal of faster release cycles.

Case study: food service distribution company

The business strategy of a food service distribution company based in Rosemont, Illinois, requires the IT department to shorten its database development cycle from two months to just a few weeks. A strategic DevOps engineer has been appointed to handle transformation of the process to DevOps.

The company already has hundreds of developers using Toad for Oracle. Some are doing manual PL/SQL unit testing, but it has been limited. The company needs to establish automated PL/SQL testing with pass/fail and code coverage reporting, as well as the ability to check source code and tests into source control. The organization is already using Bamboo continuous integration with Apache Subversion and Git source control systems.

Adding Toad DevOps Toolkit to the database development side of things will enable them to integrate PL/SQL unit testing into the organization's continuous integration process and create a DevOps pipeline that includes both database development and application development to better meet the company's business goals.

Toad DevOps Toolkit plays well with virtually any continuous integration and continuous delivery tool, including Jenkins, Bamboo and Team Foundation Server.

With Toad DevOps Toolkit, you can meet all your project deadlines across application and database deployment, while ensuring reliable, high-quality code.

CONCLUSION

Toad DevOps Toolkit enables you to integrate database development, testing and deployment tasks into your DevOps workflow. Integration is silky smooth, since Toad DevOps Toolkit plays well with virtually any continuous integration and continuous delivery tool, including Jenkins, Bamboo and Team Foundation Server.

You'll be assured by reliable, high-quality Oracle database code in your CI and CD environments. Best of all, you'll meet all your project deadlines across application and database deployment, with greater productivity and minimal disruption to your current DevOps processes.

You'll be assured by reliable, high-quality Oracle database code in your CI and CD environments. Best of all, you'll meet all your project deadlines across application and database deployment, with greater productivity and minimal disruption to your current DevOps processes.

To learn more about Toad DevOps Toolkit, please visit quest.com/products/toad-devops-toolkit or join in with the Toad DevOps Toolkit community on

Toad World at toadworld.com/products/toad-devops-toolkit.

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 the strategy and roadmap for the Toad portfolio of products worldwide. He has been with Quest Software since 2000, working in the database design, development and deployment product areas, and has run many Toad training courses for customers. John has spent over 17 years successfully evangelizing Toad to customers at various events around the world, and he writes many blogs and papers that are posted on the Toad user community, [Toad World](http://ToadWorld), as well as technical papers about Toad that are published on the Quest Software website.

John has worked in IT for more than 30 years, most of that time in Oracle application design and development. He is a qualified aeronautical engineer with more than 10 years of experience in provisioning IT consultancy services and implementing quality assurance systems to ISO 9001.

ABOUT QUEST

At Quest, our purpose is to solve complex problems with simple solutions. We accomplish this with a philosophy focused on great products, great service and an overall goal of being simple to do business with. Our vision is to deliver technology that eliminates the need to choose between efficiency and effectiveness, which means you and your organization can spend less time on IT administration and more time on business innovation.

© 2017 Quest Software Inc. ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software 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 Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE 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 SOFTWARE 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 SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software 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 Software does not make any commitment to update the information contained in this document.

Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at www.quest.com/legal

Trademarks

Quest, Toad and the Quest logo are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit www.quest.com/legal/trademark-information.aspx. All other trademarks are property of their respective owners.

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

Quest Software Inc.

Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our Web site (www.quest.com) for regional and international office information.