DEVOPS FOR ORACLE DATABASES

Bridge the gap between development and operations with Toad®

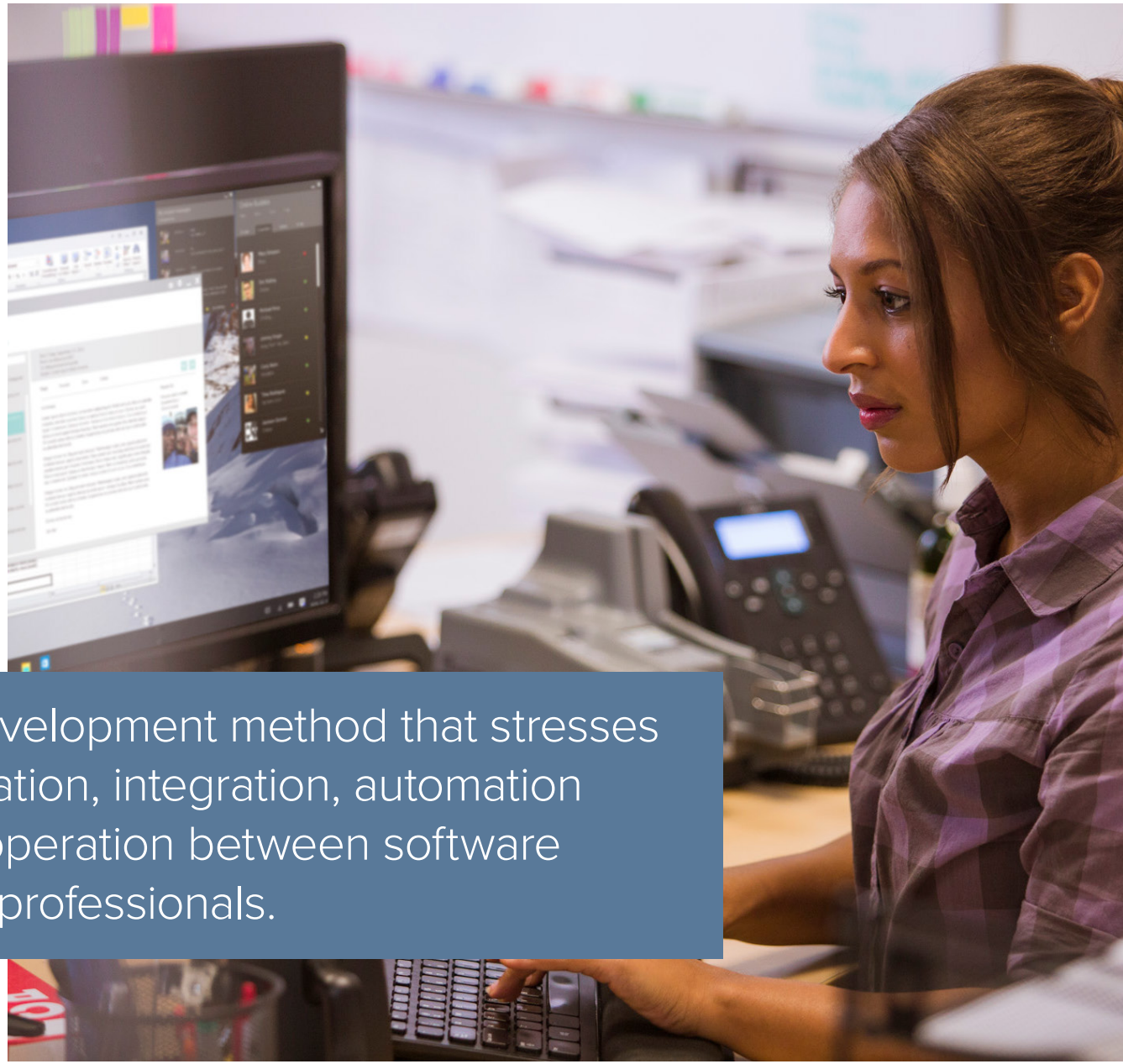
Quest®

What is DevOps?

DevOps is sometimes treated as a concept, a philosophy, a job role or a set of tools.

But most IT professionals would agree with Wikipedia's definition:

DevOps is a software development method that stresses communication, collaboration, integration, automation and measurement of cooperation between software developers and other IT professionals.



DevOps is a software development method that stresses communication, collaboration, integration, automation and measurement of cooperation between software developers and other IT professionals.

Quest

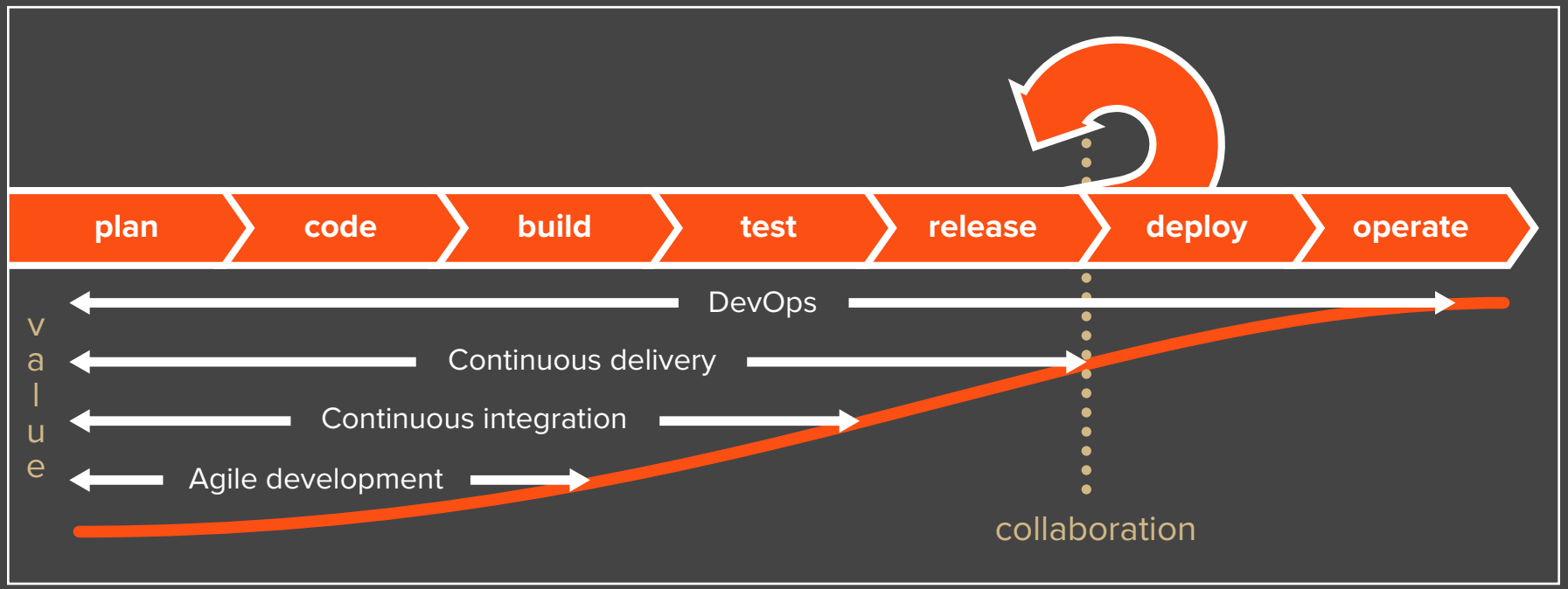


Figure 1. DevOps is a software development method that evolved from earlier agile development, continuous integration and continuous delivery initiatives. (Adapted from CollabNet)

As illustrated in Figure 1, DevOps evolved from the agile development, continuous integration and continuous delivery initiatives that began gaining traction in 2008. The goals have remained the same throughout this evolution:

- Speed time to market
- Lower costs
- Reduce risk
- Improve productivity

Quest

Why your organization needs DevOps

Your organization is struggling to keep pace with rapidly changing market requirements.

You're faced with multiple disruptive forces — mobility, cloud, big data analytics, social business and more. You're being asked to deliver software changes faster to meet the demands of rapidly changing market requirements. You know you need to integrate database change management into your DevOps culture, but simply putting stop-gap solutions in place or continuing with uncoordinated processes is not the answer. The risks of a failed implementation are too great.

Depending on traditional processes that are slow, manual and error-prone leads to long delays as development, QA and testing teams wait for each other. When an issue arises, troubleshooting is difficult, and the development teams who wrote the code and the operations teams who deploy it are often at odds, playing the blame game. What if you could accelerate your database pipeline end to end and combine higher quality database changes with application changes? What if you could reduce application release time, spend less time fixing defects and more time innovating?

DevOps can help organizations speed deployment, improve agility, reduce risk and break down silos.

What if you could do all this without compromising quality, performance or reliability?

A DevOps culture can help you:

- **Speed deployment processes —** Organizations are spending too much time with manual processes around testing and deploying software. DevOps can automate tasks to not only speed delivery but improve quality.
- **Enable fast response to changing business requirements —** Just 17 percent of IT executives believe they can deliver fast enough for the pace of business.[1] Agile organizations have 30x more frequent deployments and 8,000x faster lead times than their peers.[1]
- **Reduce the risk of production changes causing outages —** Manual processes and lack of visibility to diagnose issues quickly increase risk. A Gartner study projected that "80% of outages impacting mission-critical services will be caused by people and process issues, and more than 50% of those outages will be caused by change/configuration/release integration and hand-off issues." DevOps addresses these unplanned outages by improving ways of working, automation and continuous deployment.[2]
- **Break down silos —** By facilitating communication and collaboration, DevOps can help eliminate arguments and finger-pointing between development and operations teams, resulting in more effective troubleshooting and faster problem resolution.

1 2014 State of DevOps survey by Puppet Labs.

2 "Top Seven Considerations for Configuration Management for Virtual and Cloud Infrastructures." Gartner, Inc., October 2010.

Accelerating software delivery with DevOps database tools

How can DevOps database tools help you accelerate software delivery? By helping you with the five tasks shown in Figure 2:

1. **Manage code changes —** Version control plays a different role on database development projects than it does in application development, but it is still a valuable way of tracking data definition language (DDL) changes every step of the way. Tracking revisions to stored functions and procedures over time makes it easier to assess them, compare them side by side and diagnose any problems that arise on the path toward production.
2. **Test early and often —** The only way to be certain that a proposed change does not break something else is through automated unit and regression testing. Even if developers and QA teams are comprehensive and remember everything they need to test, manual testing is still vastly slower than automated tests that run against the code change at the point of check-in.

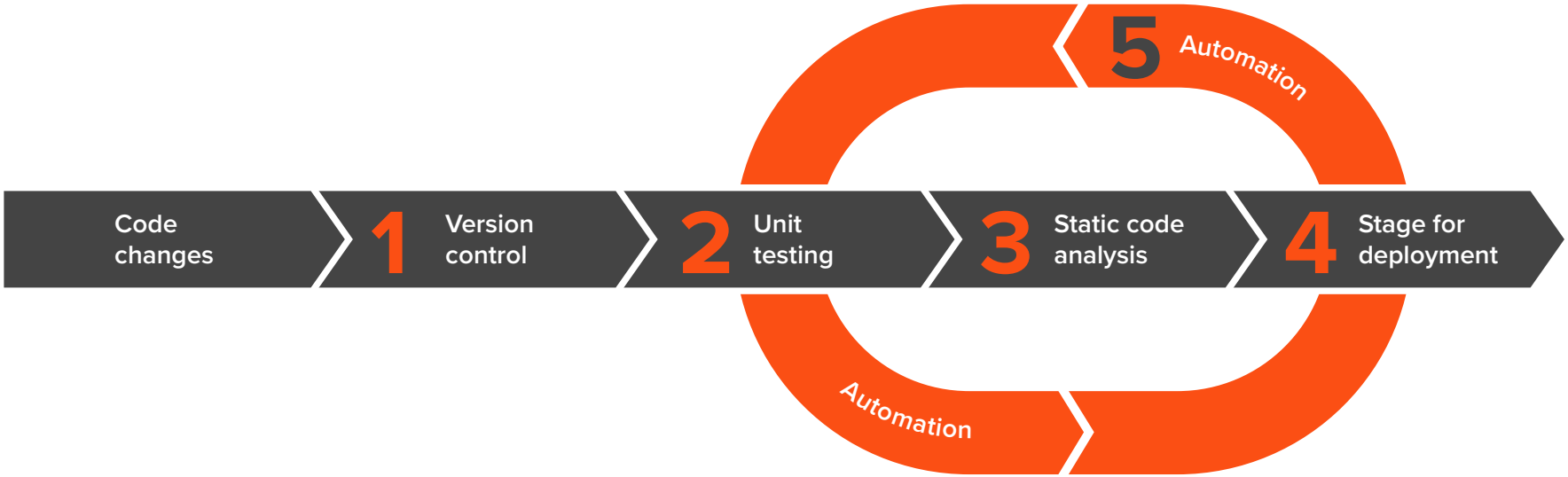


Figure 2. DevOps database tools can help you with every step of the software delivery process.

Quest

Automated testing provides a safety net for accelerating database deployments because it offers nearly immediate assurance that new changes do not break earlier work. Furthermore, by automating and executing this process at the point of check-in, developers will receive immediate feedback about a break and be in the best position to fix it quickly, when the relative cost to fix a bug is at its lowest.

3. **Ensure quality with standards —** Developers subject their code to peer review to ensure that they haven't missed something, introduced a security vulnerability, made a mistake in logic or inadvertently slowed the product down. Static code analysis software significantly accelerates this process and ensures adherence to company standards by reading the code and identifying the same kinds of patterns that peer developers look for.

4. **Deploy faster —** To reduce the risk of data loss and other mishaps, the database development path usually includes a stop in the deployment stage for the DBA to review code changes before they go into production.

   DBAs and developers managing the path to deployment can use automated tools to collect all the queued changes that have passed regression tests and static code analysis, compare them to the production environment, and generate the scripts to commit them. Not only can this improve DBA efficiency and shorten the development cycle, but it can also ensure all of a project's changes make it into production. This is also an opportunity to check the SQL code and ensure it has been optimized and check for any unforeseen issues.

Reaping the full benefit of agile requires a fundamental shift in the way teams work.

5. **Automate everything —** As the organization moves away from manual processes and toward automated tools, database development cycles will start shrinking and all teams can begin to realize the promise of agile. Using software tools in a piecemeal fashion each step of the way is faster than purely manual processes, but it provides only incremental improvements.

   Reaping the full benefit of agile requires a fundamental shift in the way teams work. Instead of having people manage each step of the process, teams must look for ways to create fully automated pipelines that submit changes to regression testing, review and staging for deployment without further interaction from the team. DBAs can rest assured that code changes meet quality standards and adhere to company policy, and managers can see that code will meet project requirements and run properly in production.

TASK 1:

Manage code changes

Toad® includes a powerful source control feature, Team Coding, that helps development teams manage code changes in the database, including preventing developers from overwriting the changes of other team members.

Figure 3 shows some of the options available to you. With Toad, you can:

- **Use your existing version control system —** Toad works with all the major version control systems, including Git, Microsoft Team Foundation Server (TFS), Concurrent Versions System (CVS), IBM ClearCase, Perforce and Apache Subversion (SVN).
- **Improve productivity —** Developers can seamlessly check code in and out directly from within Toad instead of juggling multiple tools.

Toad includes a powerful source control feature, Team Coding, that helps development teams manage code changes in the database.

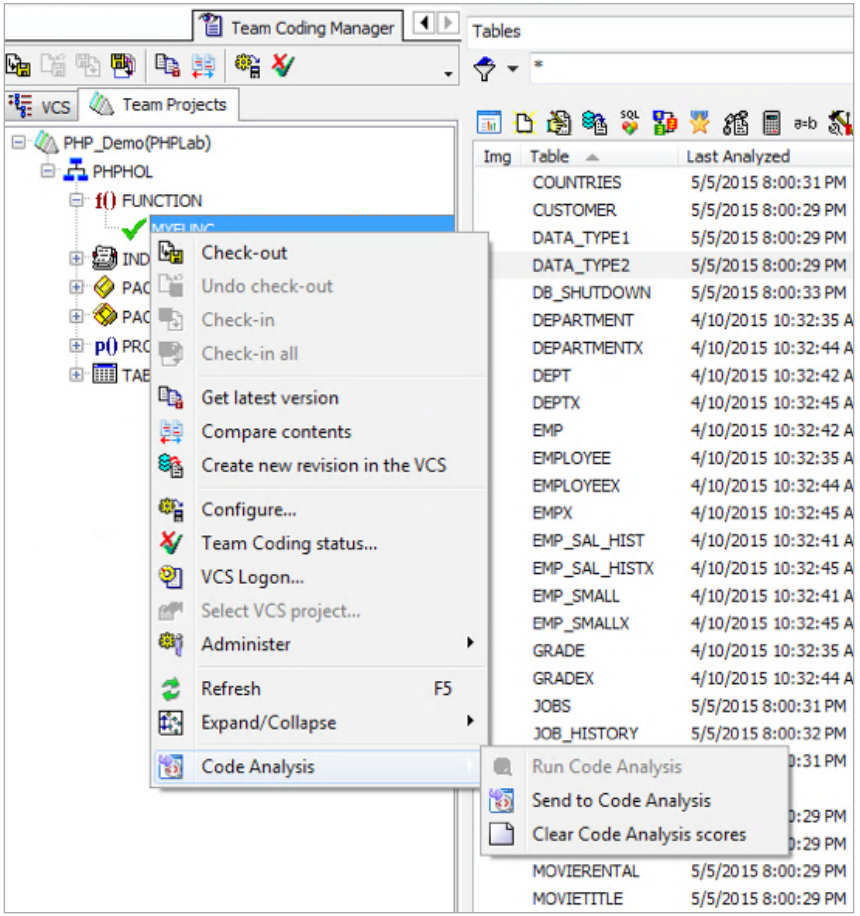


Figure 3. Seamlessly check code in or out directly from within Toad instead of juggling multiple tools.

Quest

TASK 2:

Test early and often

Toad Code Tester for Oracle automates the process of testing PL/SQL programs by identifying bugs and verifying program correctness in a fraction of time that would otherwise be required (see Figure 4).

Code Tester helps development teams:

- **Produce better code —** Because developers can create representative test cases automatically, they can test more often and more thoroughly to ensure the correct function of their code.
- **Reuse test cases —** Test cases are stored in a repository so they can be reused for later regression testing.
- **Work more productively —** Code Tester for Oracle is integrated with Toad, so developers can easily create new unit tests as a part of their debugging activity.
- **Keep test definitions in sync as code changes —** Code Tester for Oracle includes a code evolution feature that ensures unit test definitions are kept in sync with code changes.

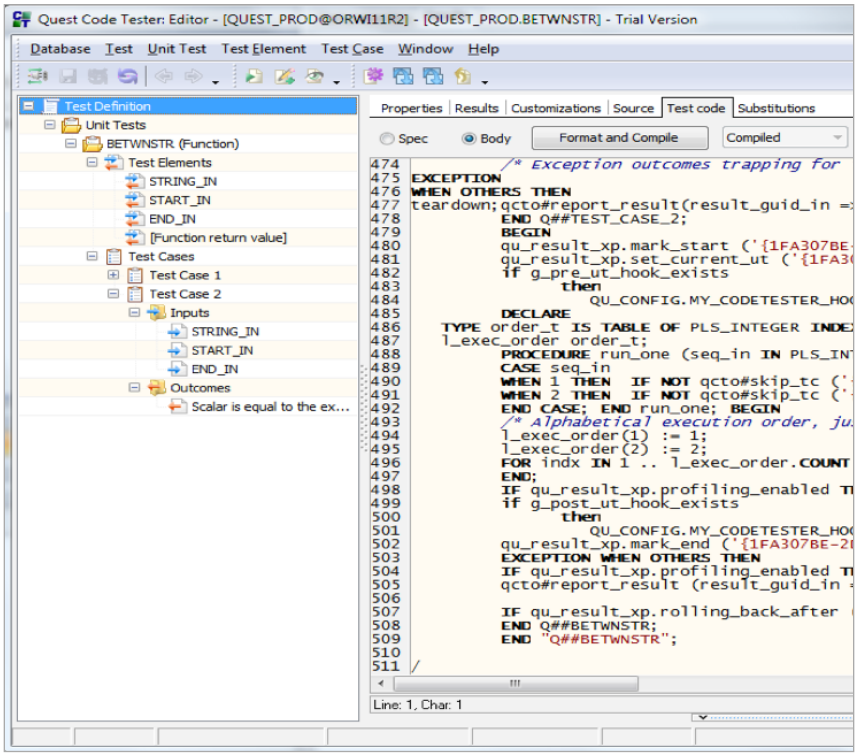


Figure 4. Toad Code Tester for Oracle automates the testing of PL/SQL programs.

Quest

TASK 3:

Ensure quality with standards

The Code Analysis feature of Toad provides automated code review based on a rules engine to determine whether code meets company standards and follows industry best practices (see Figure 5).

Code Analysis helps development teams:

- **Improve code quality and consistently follow best practices —** Code Analysis ensures that developers working in a team environment are consistent in making code changes. It also ensures that they use standard processes for making changes, eliminating backdoor methods that could admit substandard code.
- **Ensure code is production-ready —** Automated code review enables developers and managers to make more accurate decisions about code readiness, thereby reducing disruption and minimizing unplanned development cycles.
- **Improve agility —** Automation speeds the code review process, making each development cycle better and faster.
- **Gain better visibility —** A centralized dashboard and analytics on code quality provide managers with the insight they need into code quality.

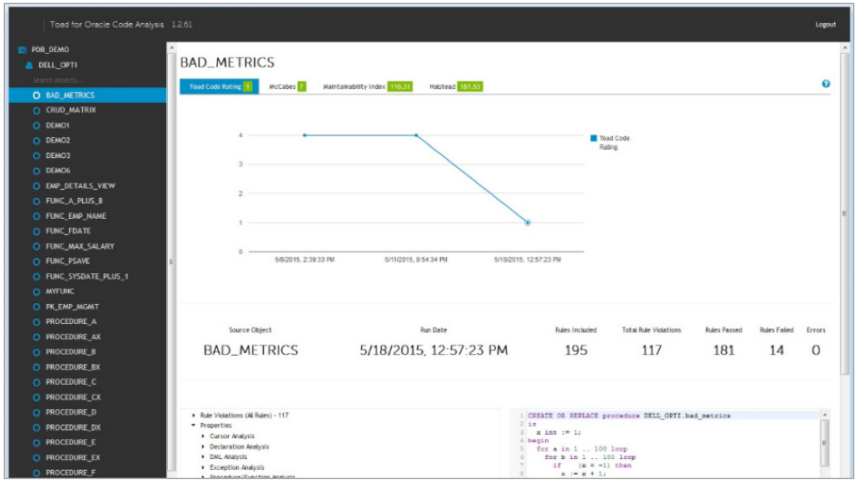


Figure 5. Automated code review in Toad provides insight into code quality.

Toad Code Analysis provides automated code review based on a rules engine.

Quest

TASK 4:

Deploy faster

COMPARE AND SYNC SCHEMAS

The task of collecting changes and writing the ALTER script to move those changes from one environment to another is tedious and highly error-prone. Automation solves both of those problems. With push-button ease, the compare and sync utility of Toad simplifies the process of generating the schema change script, saving time and reducing the chances of human error due to improper sequencing or forgotten changes (see Figure 6).

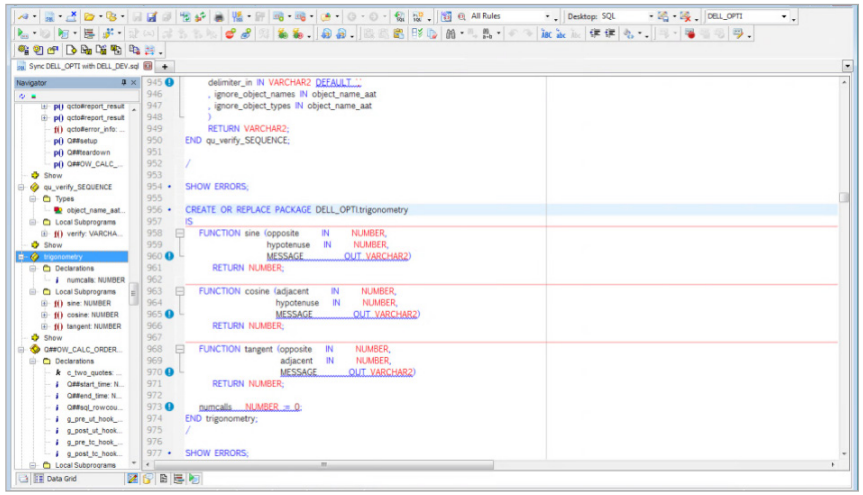


Figure 6. Toad simplifies and automates schema comparisons.

PROACTIVELY TUNE SQL

Even high-quality code may run into performance issues inside the database. Rather than have DBAs try to tune SQL in a post-production environment when performance is suffering, developers can use the SQL-tuning capabilities of Toad to proactively and automatically identify the best way for their SQL to function at peak performance (see Figure 7).

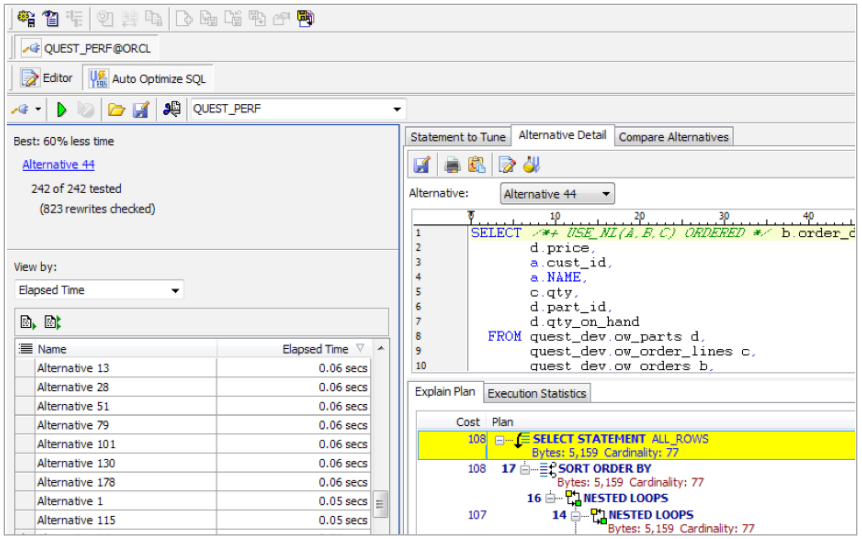


Figure 7. Toad automatically identifies problematic SQL so developers can proactively tune it, preventing costly problems in production.

Quest® SQL Optimizer for Oracle offers even more advanced tuning options. Rather than tuning a single piece of code, you can optimize batches of code in a single operation. For instance, you can determine which index combinations would work best across the entire batch. Due to variations in the optimizer, SQL Optimizer's Impact Analysis feature can quickly identify code that may regress in the production environment.

Quest

Benchmark Factory is integrated with Toad so developers can easily check whether their code will scale properly for execution by multiple concurrent users.

ENSURE SCALABILITY

Even code that has gone through quality and tuning checks can sometimes bog down when executed simultaneously by multiple users. Benchmark Factory® is integrated with Toad so developers can easily check whether their code will scale properly for execution by multiple concurrent users.

Quickly identify, diagnose and resolve issues

Before deploying code into a production environment, you can analyze the workload in the test environment using the Spotlight® on Oracle diagnostic tool (see Figure 8). Spotlight represents the Oracle database architecture in a graphical fashion so you can quickly spot emerging issues. Moreover, Spotlight provides context-sensitive guidance that enables DevOps teams to quickly investigate and resolve problems. And should there be any performance issues after the deployment, Spotlight can help you tackle them as well.

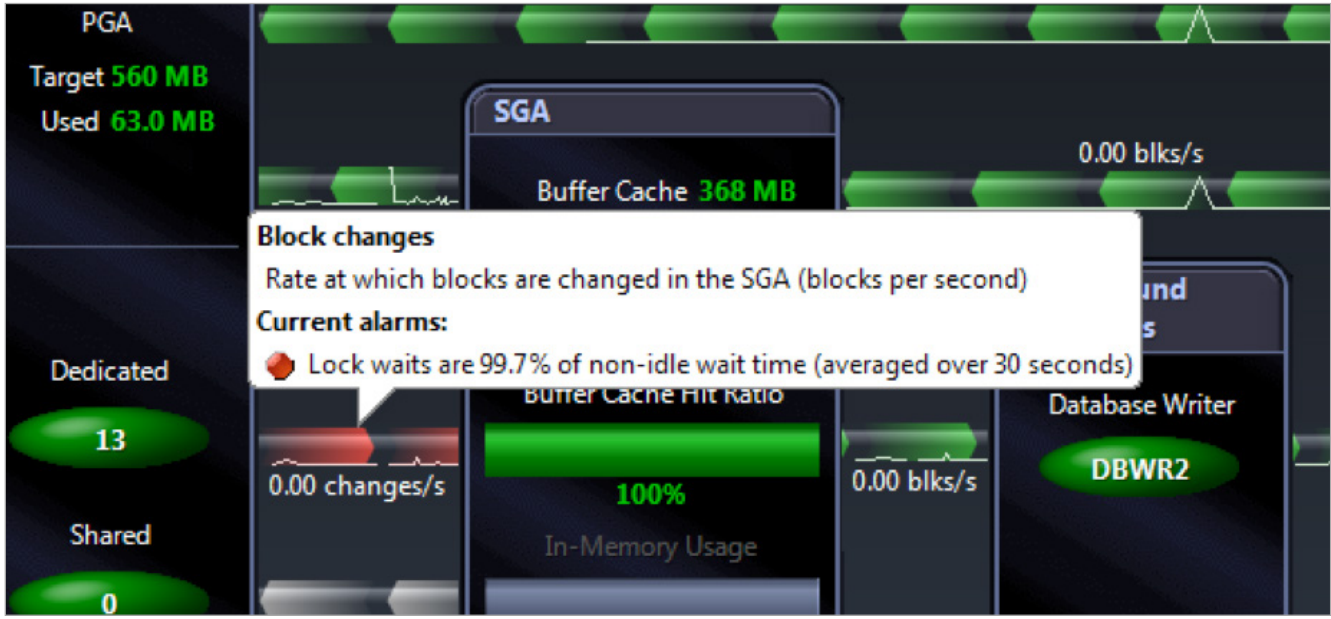


Figure 8. Spotlight alerts you to problems in the test environment and in production.

Quest

TASK 5:

Automate everything

Integrating database tasks, such as those in steps 2, 3 and 4, with established DevOps automation tools such as Jenkins and Bamboo is absolutely essential to converging application and database into the DevOps pipeline.

Toad DevOps Toolkit integrates with many build automation tools so you can include database tasks into the pipeline to shorten release cycles without compromising quality and performance (see Figure 9).

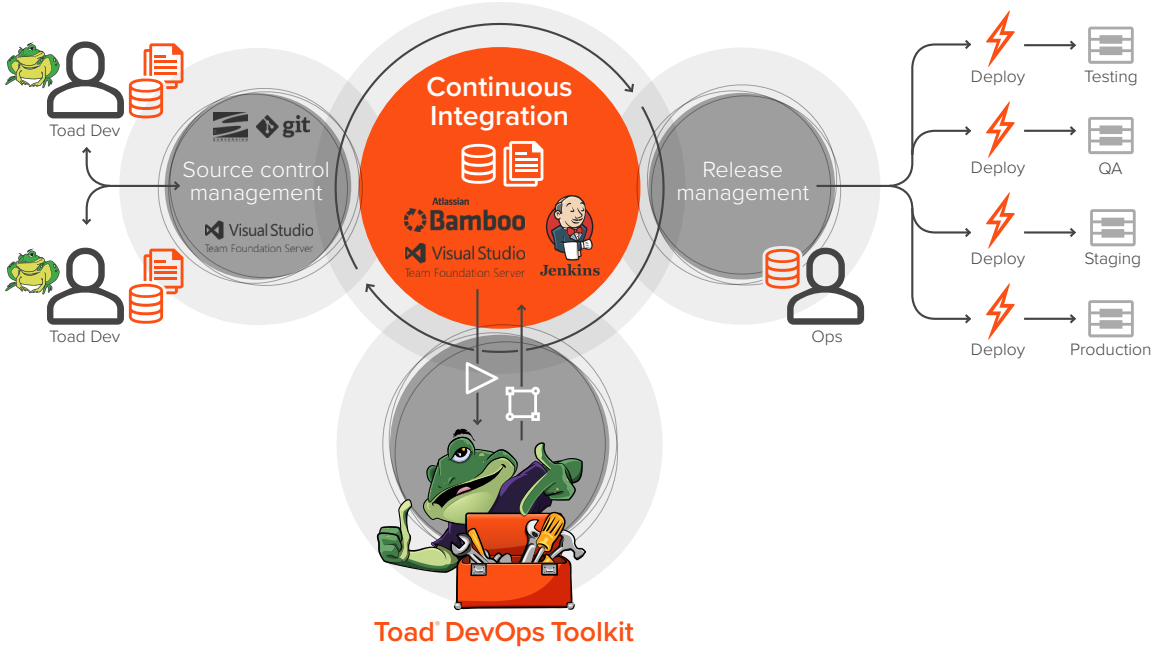


Figure 9. Toad enables you to easily automate a variety of tasks.

Automating frequent and repetitive tasks using Toad reduces the likelihood of human error and greatly improves productivity.

Quest

Learn more about DevOps and Toad

DO YOU NEED DEVOPS?

As you continue to explore whether DevOps can help your organization, ask yourself the following questions:

- Is upper management pressuring you to accelerate your delivery time and processes?
- Do you have long delays in development, QA and delivery?
- Do you have a lot of manual processes in your application lifecycle that slow down delivery time?

Toad DevOps Toolkit can help you effectively implement DevOps and thereby speed time to market, lower costs, reduce risk and improve productivity.

TOAD DEVOPS TOOLKIT CAN HELP

If you answered yes to any of these questions, Toad can help.

DevOps has been increasing in popularity dramatically over the past few years, and Quest is here to help you integrate database development into the DevOps pipeline. Toad DevOps Toolkit allows you to leverage the power of Toad during an automated build by providing access to key database development functions via programmable objects. Perform:

- PL/SQL unit testing
- Static code review
- Database compare
- Schema compare
- Data compare
- Script execution

Check out Toad DevOps Toolkit — quest.com/products/toad-devops-toolkit

We also offer a variety of resources to help you continue your broader investigation of DevOps and agile methodologies, including white papers, tech briefs, and on-demand webinars.

Visit quest.com/database-management for more.

Quest

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.

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 website (www.quest.com) for regional and international office information.

Ebook-DevOpsForOracleDB-US-KS-36807

Quest