NeoDevel (Web)

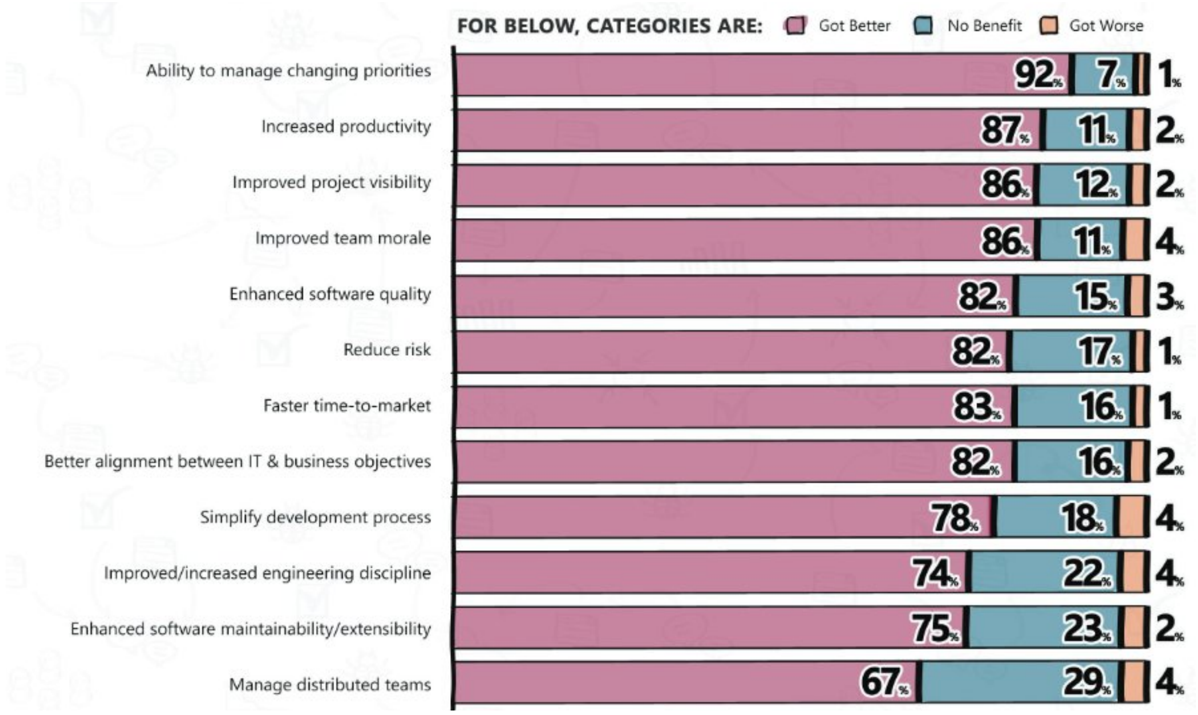
Agile Software Framework (DevOps):

Quality and reliability of a manufacturing line applied to software development

Today's businesses are complex organizations that must be agile across highly competitive global markets. To succeed, their IT departments must work as efficiently and reliably as a manufacturing assembly line, where each stage of development and release process is clearly defined and automated to minimize errors and delays. To help businesses address these needs, Neoscopio offers Agile Software Framework (NeoDevel) – a Continuous Delivery toolkit that uses pre-integrated tools to implement end-to-end development and release processes. Industry standards and best practices implemented in NeoDevel serve as a solid foundation for assuring an efficient, agile (SCRUM, XP,KANBAN etc), and structured software development culture. Offering a wide range of standard pre-configured features, NeoDevel can be easily customized to address unique requirements and integrate with a business existing processes and tools.

Introduction

The fast-changing landscape of modern business makes it extremely challenging for companies to maintain their competitor edge and market leadership. Customers and other stake holders are more technically savvy than ever before and expect and demand an experience customized to his or her preferences. The added complexity of supporting multiple channels, coupled with the increasing size of engineering teams, makes it more difficult to deliver the necessary software quickly and reliably. To succeed, companies need to take a more integrated and structured approach to their software development and integration, built on well-defined processes, standards, and best practices.



Top benefits from adapting Agile methods – 2013, 8th annual state of Agile survey

Industry overview

Not so long ago, building and releasing software involved relatively small teams that could effectively use informal processes involving ad hoc procedures and manual hand-offs In contrast, today's modern software engineering organizations tend to require the coordination of many stakeholders composed of several inter-department developers, QAs, architects, analysts, managers, and release engineers. As a consequence, software development processes have become much more complex, leading to longer release cycles and greater risk.

Over the past several years, companies have had to make significant investment in their software ERP platforms continuously adapting to stay relevant in an increasingly competitive marketplace. Such investments lead to tremendous growth of IT departments, demanding fast and efficient innovation. Technology giants like Amazon, who implemented Continuous Delivery through many years of trial and error, are now able to deploy to production at blazing speed and with greatly reduced risk. Traditional companies, in contrast, don't have the luxury of spending a lot of time to set up proper processes. Their IT departments need to achieve better efficiency from day one.

problem description

With the increasing growth of engineering teams, businesses find themselves facing new kinds of software delivery issues:

- Releases are constantly delayed due to either critical defects appearing in late design stages or lack of agreement about what functionality should go to production.

- Management doesn't have a clear picture of the completion status of each feature planned for release. This increases overall uncertainty and delivery risks.

- Defects take longer to fix due to lengthened feedback cycles. For example, because testing is often completed just before release, developers have lost the context of any defects found.

- Development environments have become complex and difficult to set up, and development and QA teams lose valuable time in installing setting up and upgrading Operating systems, middle layer software and even programming language. Specifically, on web development, special attention must be given to constantly changing browser versions/technology.

Tools integration

The value of Continuous Integration / Delivery and a structured development culture is generally recognized today. Businesses have a wide choice of tools to help with implementing very specific aspects of Continuous Delivery, including source code management, code quality analysis, deployment automation, environment management, and testing automation, along with a large vast choice of special propose tools. However, having to select the optimal combination of tools and integrate them together makes the challenge of implementing a Continuous Delivery-based infrastructure a hard endeavor.

Even when a business can overcome these technological difficulties, its teams implementing Continuous Delivery often need to invent custom processes from scratch. Not many companies can afford to go through the long experimentation cycles of trial and error required to assure these processes are robust enough to trust critical business operations to. Moreover, such implementations often result in the creation of specialized development and release process knowledge that typically belongs to a small number of key individuals. This exposes businesses to the risk of these individuals leaving the company and taking their knowledge with them.

NeoDevel: Key features

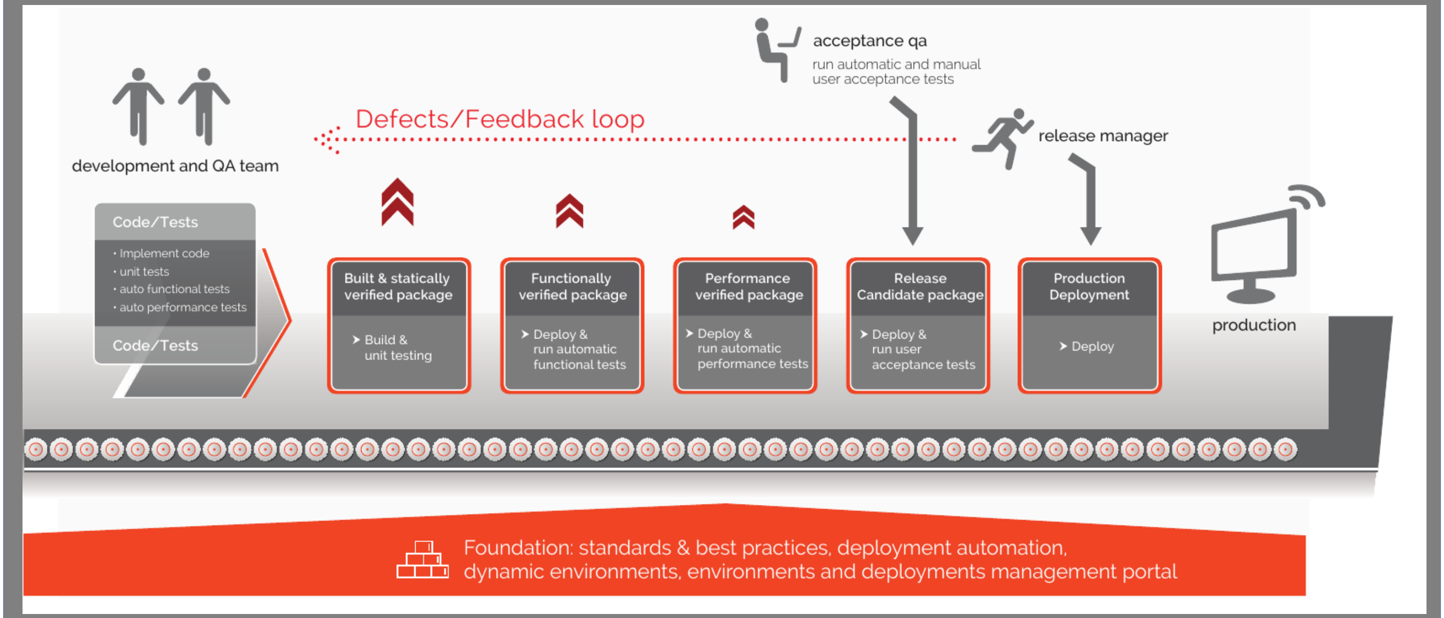
To help companies address these challenges, Neoscopio offers Agile Software Framework (NeoDevel). Agile Software Framework provides a ready-to-use implementation of Continuous Delivery out of the box through self-deployable, integrated components that create well-defined development and release processes, standards, and best practices.

Among the many benefits Agile Software Framework can bring to businesses are:

- **Speed:** NeoDevel increases the rate and frequency with which new features can be implemented. Deployments that used to take days can be completed consistently and reliably in less than an hour through complete deployment automation.
- **Consistency:** NeoDevel implements proven, standardized processes built upon well-documented standards, application properties management, configuration management, and data management. This ensures not only correct usage of ASF tool chain but also establishes efficient development and release processes and culture.
- **Visibility:** A centralized project interface for management, deployment, and reporting (redmine) allows real-time visibility of the development process. Role based information provides each stake-holder with relevant information for everyone, Product owners, development teams, and even product users.

Release early, release often

The key is automation – automated testing allows for small features to be integrated faster and safer, allowing development to react faster to user feedback realizing, the mantra “Release early, release often”


acceptance qa
run automatic and manual user acceptance tests
Defects/Feedback loop
release manager
development and QA team
Code/Tests
· Implement code
· unit tests
· auto functional tests
· auto performance tests
Code/Tests
Built & statically verified package
› Build & unit testing
Functionally verified package
› Deploy & run automatic functional tests
Performance verified package
› Deploy & run automatic performance tests
Release Candidate package
› Deploy & run user acceptance tests
Production Deployment
› Deploy
production
Foundation: standards & best practices, deployment automation, dynamic environments, environments and deployments management portal

- **Flexibility:** Dynamic management of cloud-based resources leads to time and cost savings by creating/destroying fully configured environments on demand, for branching / testing.
- **Automation:** Automated setup of pre-integrated NeoDevel components saves time on initial implementation. Components integration also ensures efficiency of end-to-end processes.

NeoDevel processes and best practices

With its integrated components, NeoDevel creates a balanced system that implements various best practices and supports customization to meet a business specific needs. The Continuous Delivery process that NeoDevel implements out of the box is shown in Figure 2.

NeoDevel implements an assembly pipeline that manages code changes to production deployments reliably and quickly, so ideas transform into revenue faster. Proper functionality of the ASF assembly pipeline is ensured by industry-proven best practices:

- **Continuous Integration (CI):** CI is the core of NeoDevel. All components and communications between components are built to provide effective continuous integration. Specifically, components implement automated testing and deployment combined into effective pipelines that makes frequent integration of code changes possible while enabling large developers teams to work more effectively and deliver functionality faster.
- **Code change management:** Effective code change management is the cornerstone of every Continuous Delivery infrastructure. It is very important to keep code change management simple but keeping flexibility to promote code reviewing, documentation, special testing, or approval on a feature-by-feature basis.
- **End-to-end visibility:** Visibility into feature status is one of the critical parameters of the Continuous Delivery infrastructure, as is immediate access to test results by developers. NeoDevel components – like the project management and task tracking system, CI server, and code review server – are integrated to provide high visibility into feature status across the entire development, testing, and release lifecycle.
- **Versioning:** NeoDevel implements versioned storage of code and other objects. Every successful build has a corresponding set of code/objects stored and versioned.
- **Continuous testing:** Testing is based on test pyramid concepts. To ensure an effective CI Pipeline, NeoDevel enforces automation of Unit, Functional and Regression testing using industry standard frameworks. Test automation also significantly reduces the time required on beta stages.
- **Environment management:** Automated environment provisioning makes it possible to create new environments on demand and destroy environments on schedule. This practice leads to effective cost management of cloud-based environments.
- **Configuration as code:** It is very important to manage dependencies between all changeable assets across a project. The tools used to manage dependencies – including deployment automation scripts, CI server jobs configuration, DB configuration, application configuration, and source code – are therefore stored and versioned. This enables NeoDevel to provide a consistent set of assets at every point of time, a factor critical for reliable Continuous Delivery.
- **Application properties management:** Proper application properties management enables effective deployment to various kinds of environments. This practice streamlines the application deployment path from the development environment through test environments onto production. NeoDevel implements a complete solution to application properties management, allowing for customization of properties for every environment type or specific environment.

- **Incremental code review:** Every completed change of code should be reviewed. Incremental code review illuminates code issues at earlier development stages. This speeds deployment as well as identifies design defects earlier. NeoDevel can be configured so that every single commit in goes through the code review procedure, thus continuously improving code quality.

NeoDevel architecture and infrastructure

The NeoDevel architecture is designed to support components integration and best practices. Best practices are implemented via tight integration of tools across all the layers (see Figure 3).
To provide the infrastructure for all necessary services and best practices, NeoDevel contains a number of thoughtfully selected components. NeoDevel uses only best-of-breed Open Source components to ensure overall system reliability. NeoDevel is built on top of widely adopted, industry standard components that have been proven invaluable in different aspects of Continuous Delivery.

All tools used in NeoDevel are open source, thus significantly reducing the total cost of ownership (TCO) of the infrastructure. The list of major components includes:

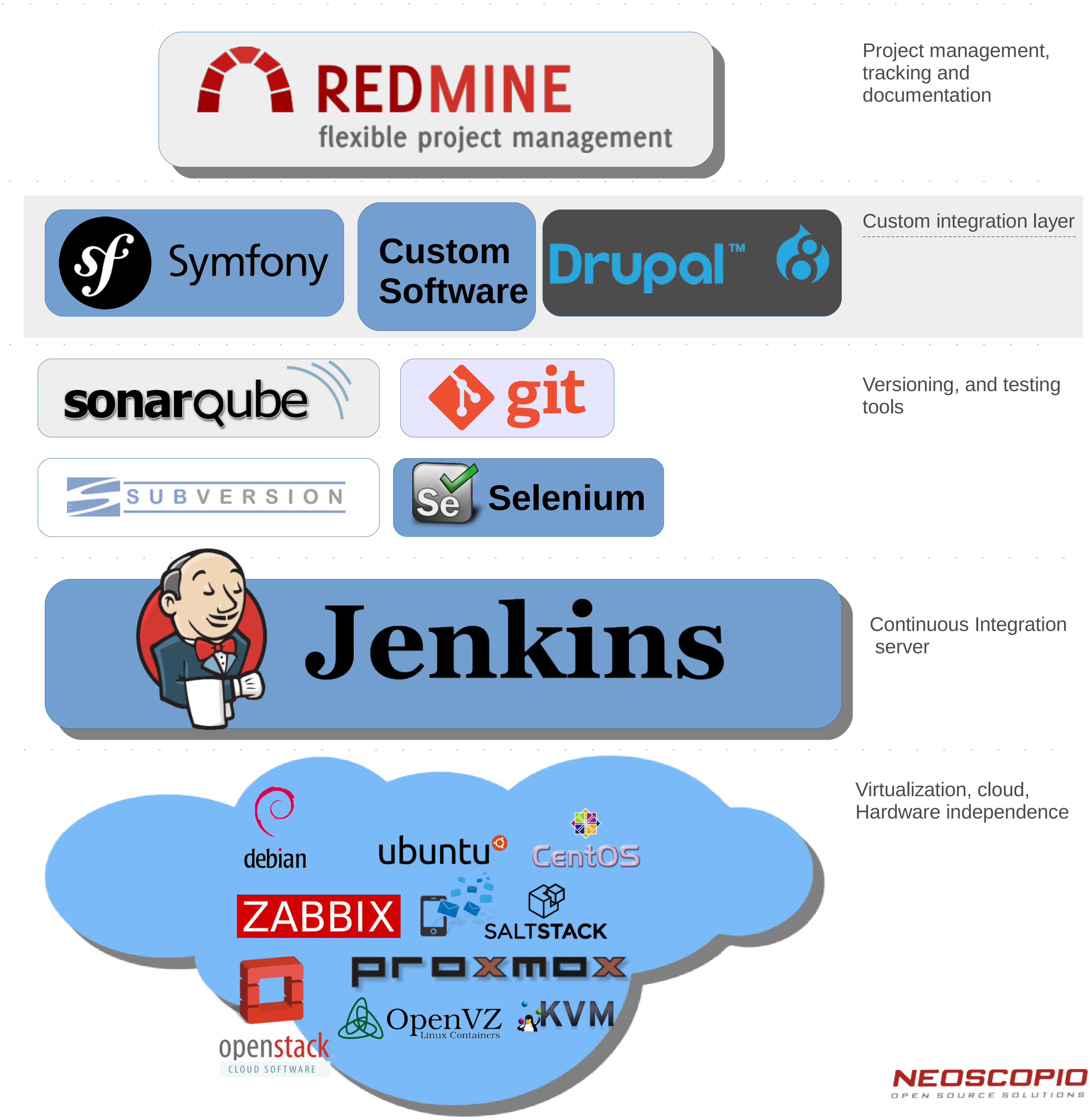
- **Redmine:** Project management, tasks, and issues tracker, with complete set of plugins to adapt to every development paradigm
- **Subversion:** Repository manager that supports binary objects storage and versioning
- **Jenkins CI:** Extendable, continuous integration server. As a pipeline orchestrator, Jenkins automates continuous integration and continuous delivery pipelines consisting of build, deploy, and test Stages.
- **SonarQube:** Open platform for managing automatic code analyses focusing on code Quality.
- **Selenium Grid:** User Interface (UI) testing automation tool. Test automation makes comprehensive CI-based processes possible and enables acceptance test-driven development.
- **ProxMox/OpenStack:** Control your own or third-party datacenter resources. NeoDevel virtualization solution provides an easy way to total solution control and hardware independence

Training:
Adapted training for custom solutions

Having a technical solution is generally not enough.
In the core of the Agile methodology, and one of the best valued features is adaptation. We provide training on Agile principles and methodology, providing IT teams with the necessary knowledge to understand and adapt to future requirements. After a specific solution for each particular case is discussed, we go a step further to provide practical training on how use the chosen tools within the agile framework practical implementation.
Specifically, our training is divided in three major areas:

1. Agile development principles, vocabulary, and methodology
2. Agile practical implementation (custom)
3. NeoDevel Framework Tools usage within practical implementation (custom)

NeoDevel Framework tools overview:



Project management, tracking and documentation

Custom integration layer

Versioning, and testing tools

Continuous Integration server

Virtualization, cloud, Hardware independence

NEOSCOPIO
OPEN SOURCE SOLUTIONS

NeoDevel components integration

To implement all of the best practices listed above, NeoDevel components are integrated into a cohesive system. This integration also supports a consistent user experience and data representation across all tools. For example, while browsing a Redmine ticket, a developer can see the links to the Jenkins build history. All Redmine tasks, and Jenkins builds have consistent information. Redmine is the Business side primary interface, and Jenkins is the technical core element of NeoDevel infrastructure. All of the other components are integrated with Redmine and Jenkins.

Ways to customize NeoDevel

Best practices implemented in NeoDevel can be used as a starting point for further customization. NeoDevel components such as Subversion, Jenkins, and Redmine support customization of environment management, how pipelines and process are establishment, and tasks and issues tracking. Build, deploy, and test run Jenkins jobs can be adjusted to particular customer projects. Environment management can be adjusted according to the preferred infrastructure as well.

NeoDevel can be also customized to work with the tools a company has already implemented and is using. NeoDevel can be customized to support Bamboo for CI; GitHub, GitLab, or any other solution for source control; Ansible or Puppet or deployment automation; and GoGrid, VMWare, or other solutions for cloud infrastructure.

Conclusion

The key to success – and survival – in today's quickly evolving market is agility. Being able to deliver value quickly, reliably, and repeated is essential if businesses are to keep pace with technology. By bringing the reliability of a manufacturing pipe to software development, businesses can have visibility into design processes, avoid undesirable delays, and eliminate defects early in the design cycle when they are less expensive to resolve.

By building a structured development culture founded on Continuous Delivery, Agile Software Framework brings consistency, flexibility, and automation to software design while accelerating the rate at which businesses can deliver new features and act on market opportunities.

With tight integration of tools across all layers of the development process, Agile Software Framework provides a sound foundation upon which companies can bring their value to market, reduce TCO, and transform the way they do business.

About Neoscopio

Neoscopio is a leading provider of open, scalable, next-generation Open Source technology solutions. With in-depth expertise in Open Source Software quality evaluation, and development, wide involvement in the open source community, Neoscopio helps great companies gain a sustainable business advantage by implementing and managing solutions in critical areas where continuous delivery and fast adaptation is valued as key factor. To learn more about Neoscopio, find us at www.neoscopio.com.