JavaOne™

ORACLE®

Developer Cloud Service – An Overview

Product Mangement – Oracle Development Tools
October, 2015

JavaOne ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Program Agenda

1. Challenges Facing Development Organizations
2. Developer Cloud Service Overview
3. Demo
4. Developer Cloud Service – A Use Case

Why Move to the Cloud?

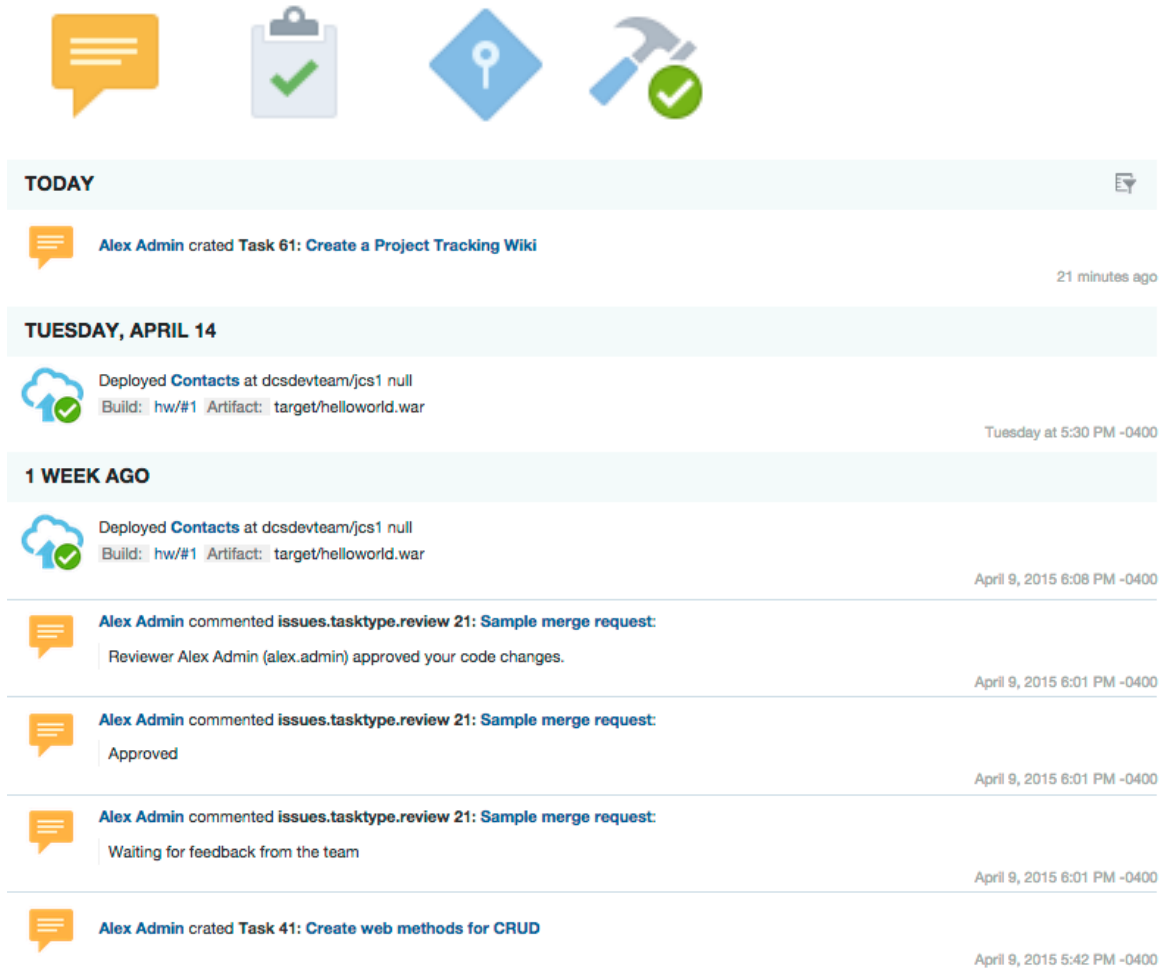
Better applications
developed faster
cheaper

Development Organization Challenges - Costs

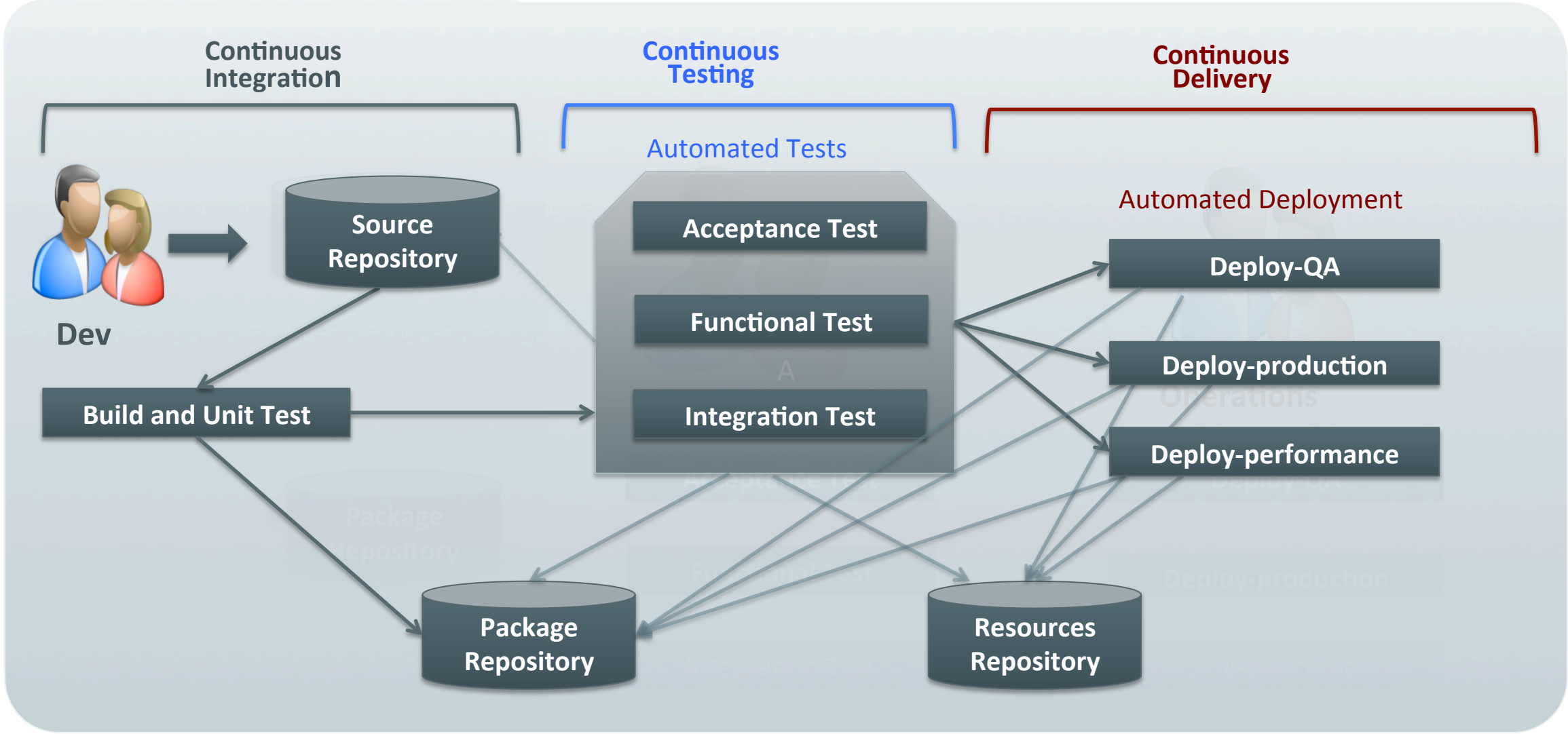
- Acquire hardware and software
- Setup and install components
- Connect components
- Configure IDEs
- Maintain and upgrade versions
- Connect to deployment platform

Development Organization Challenges - Process

- Achieving build process agility
- Producing better applications
- Streamlining deployment
- Managing teams and developers
  - Tracking and reporting
  - Team communication
  - Workload management and prioritization



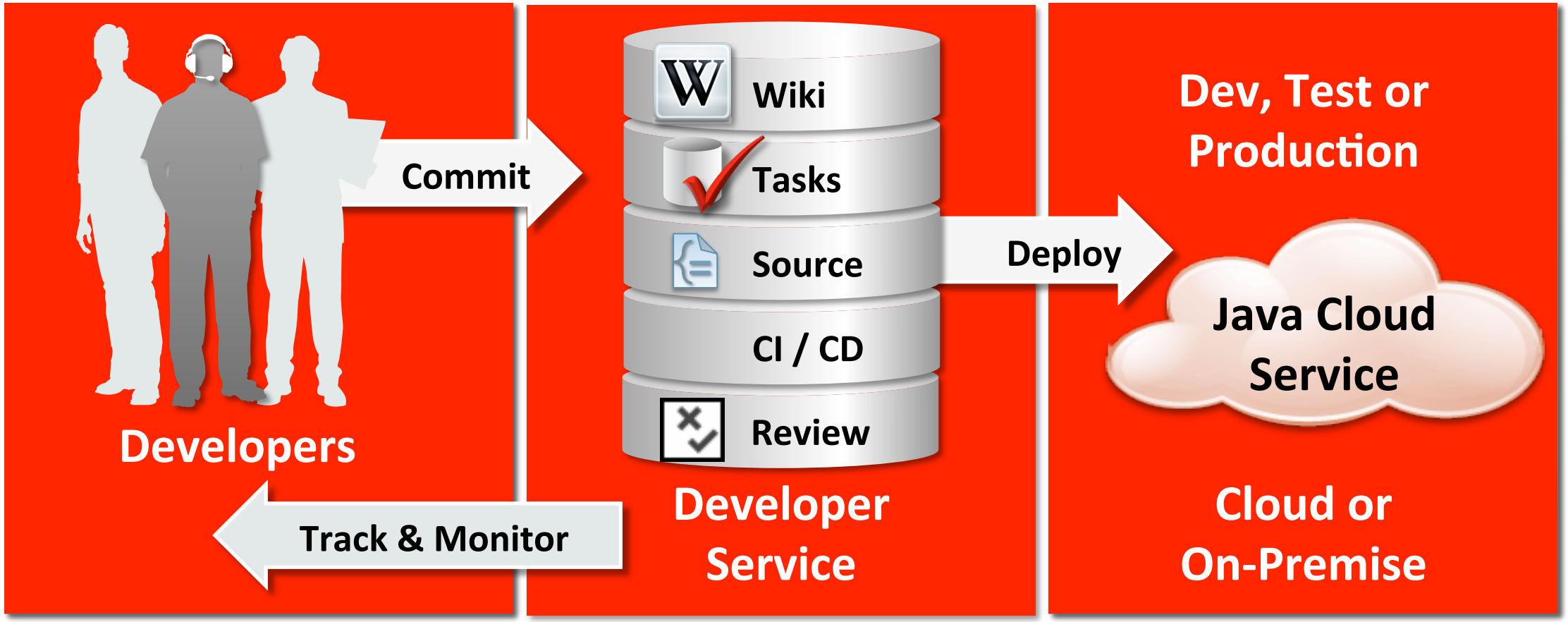
Modern DevOps



The Ideal Solution

- Integrated DevOps solution
- Quick startup time and easy provisioning
- Minimal maintenance costs
- Informative tracking of development activities
- Seamless deployment
- Flexible - cloud or on-premise

Developer Cloud Service: Bringing it All Together



Developer Cloud Service: What It Is

- Development Platform provided as a Service
- Application Lifecycle Management
- Team Collaboration & Management
- Delivery Management

Source Control Management

Issue Tracking

Continuous Integration

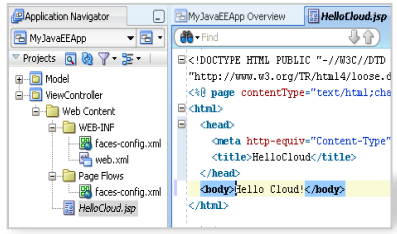
Wiki Collaboration

bit.ly/HOL10381

cloud.oracle.com

Launch Eclipse

Developer Cloud Service - Interfaces



Mylyn

JDeveloper, NetBeans and Eclipse

Web Dashboard

REST Interface

SSH to GIT

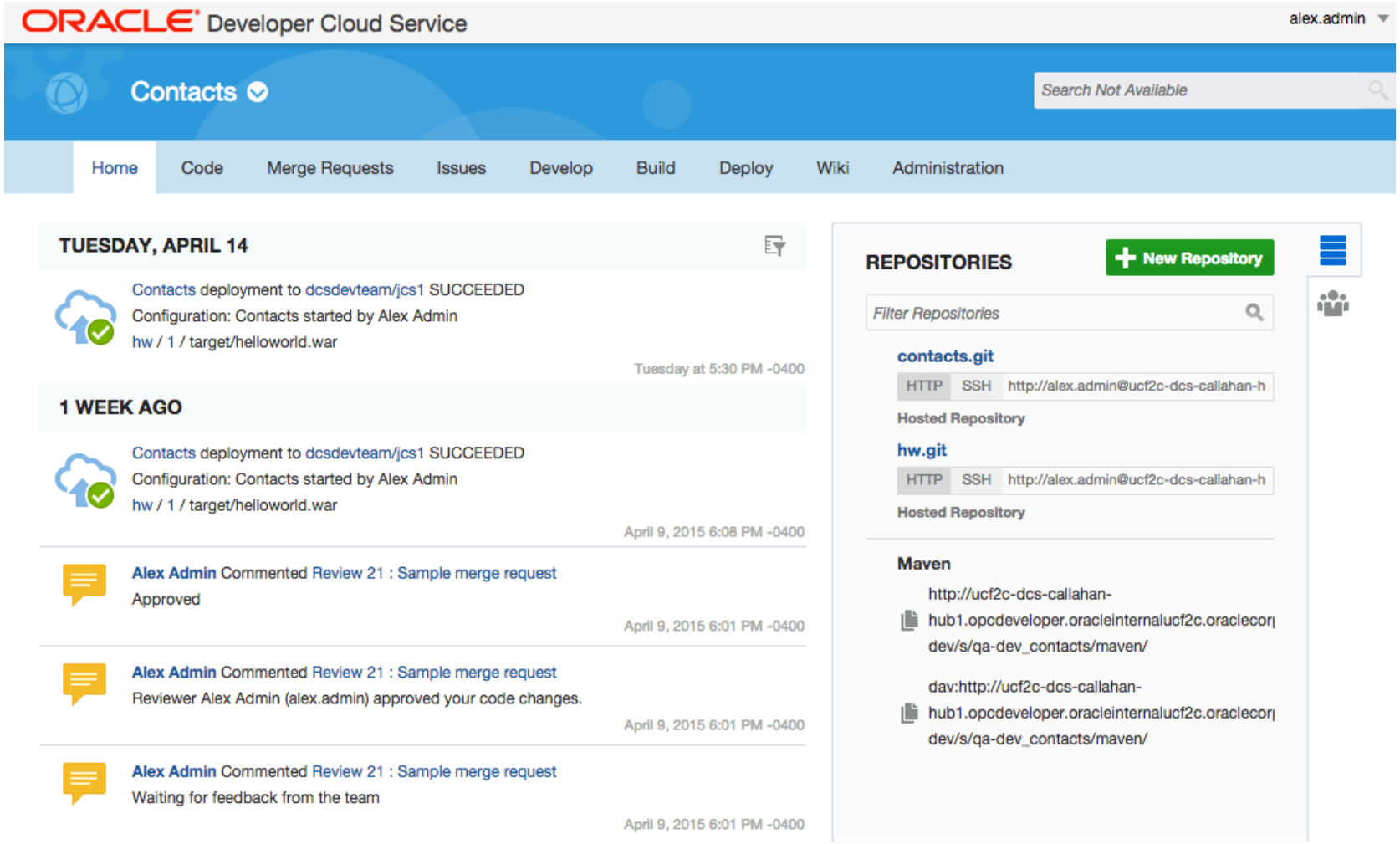
Mylyn

Partners

Developer Cloud Service

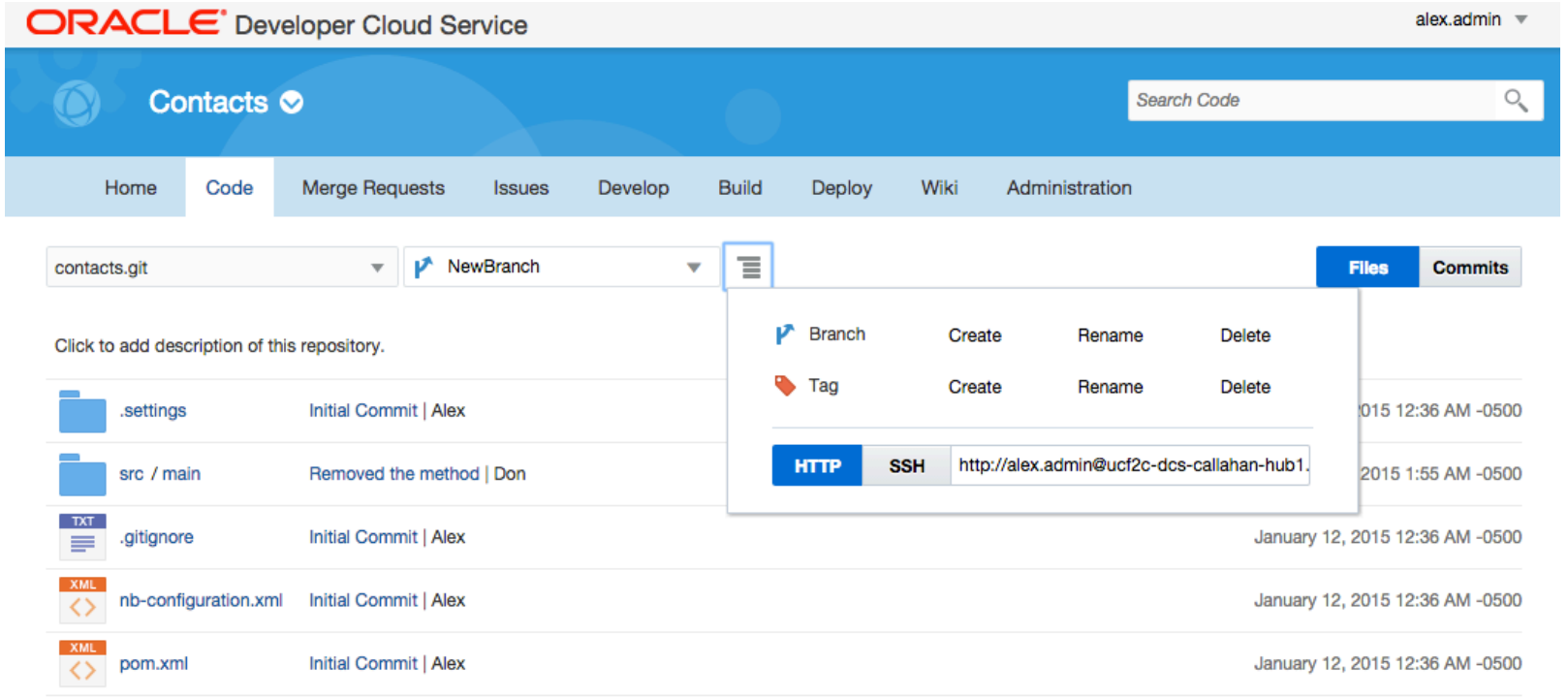
Manage Your Projects

- Activity Stream
- Git Repositories
- Maven Repository
- Team Members



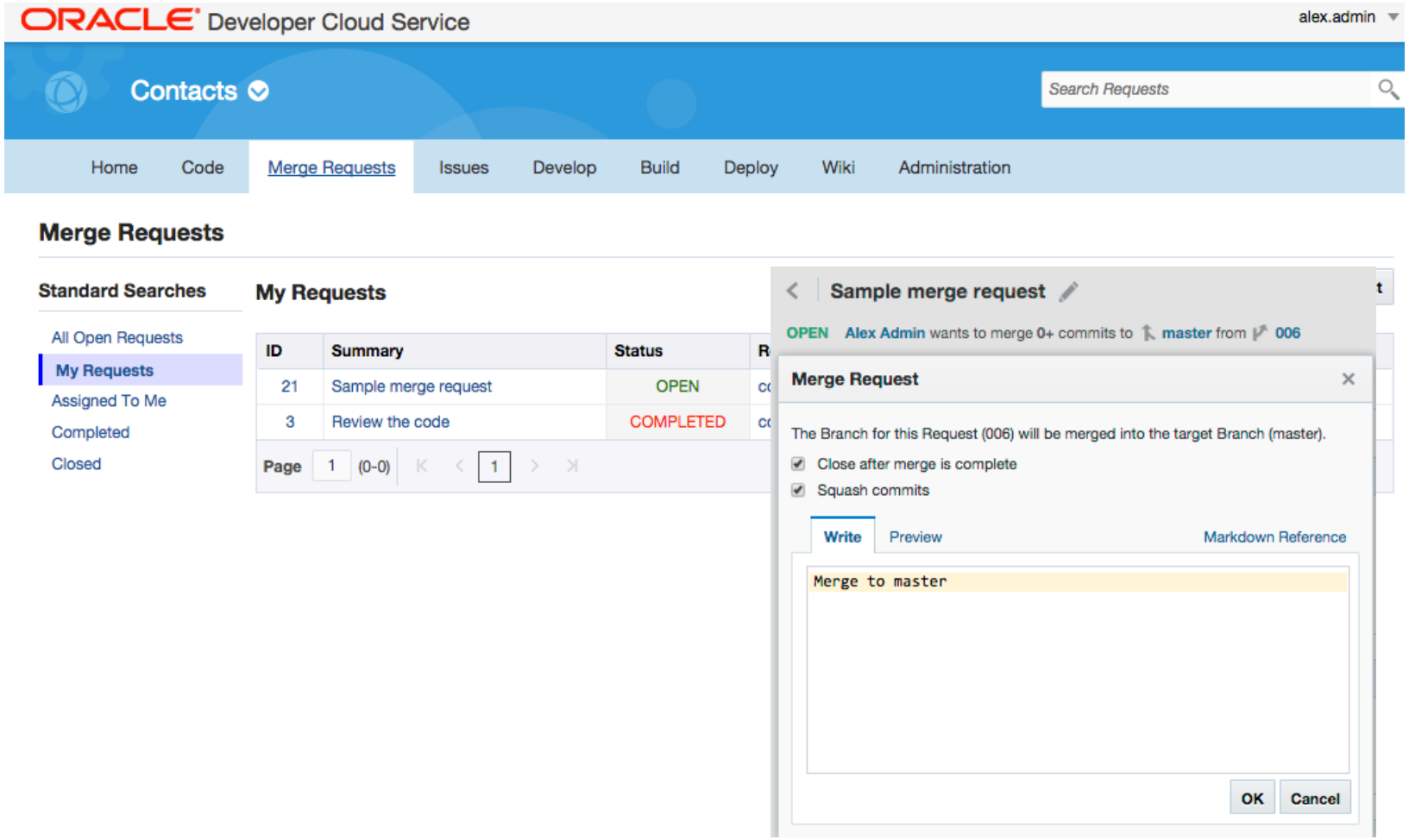
Version Your Code With Git

- Automatically provisioned
- Connect from any IDE
- Command line accessible
- Integrate with GitHub



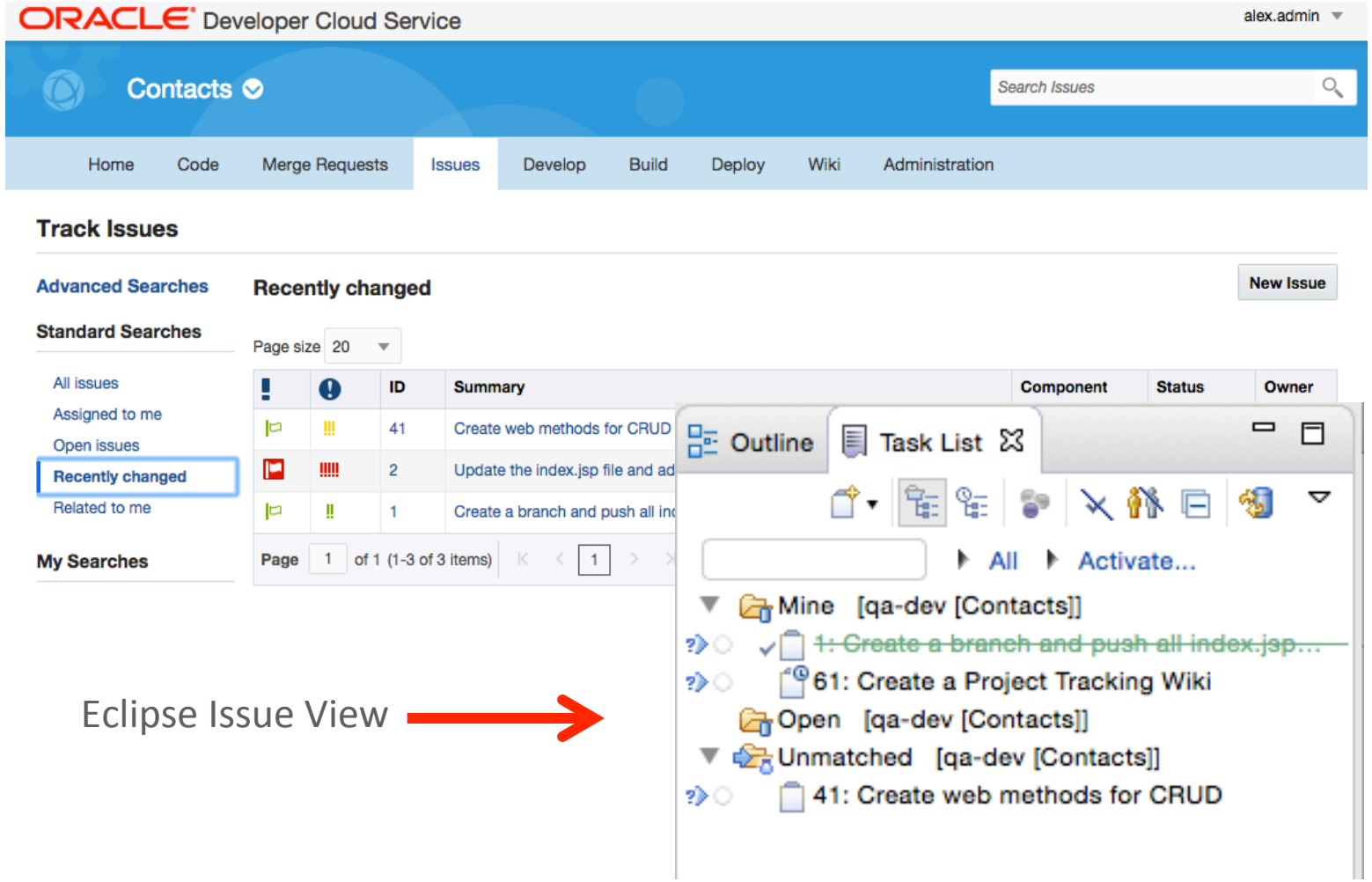
Review Peers Code

- Create Code Reviews
- Invite Team Members
- Collaborate on Reviews
- Accept / Reject / Iterate Reviews
- Comment on Code
- Merge Code
- Merge Conflict Resolution



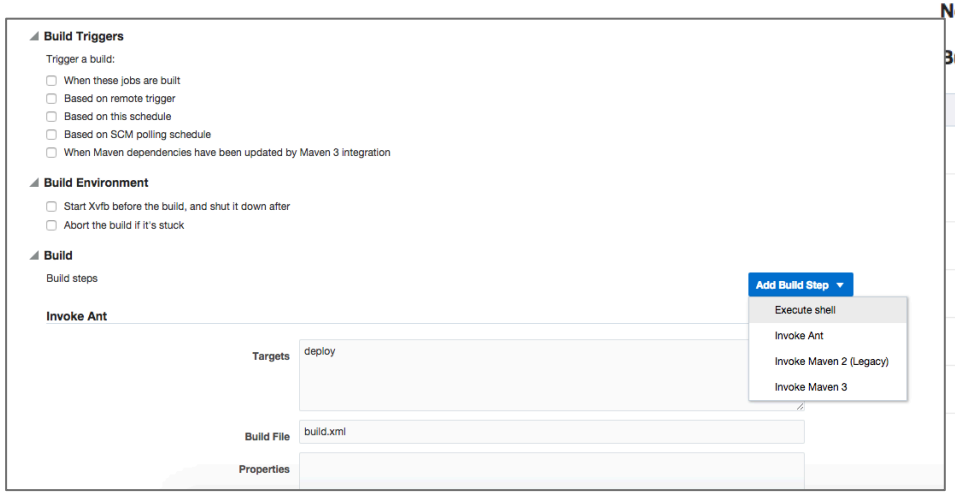
Track Project Issues

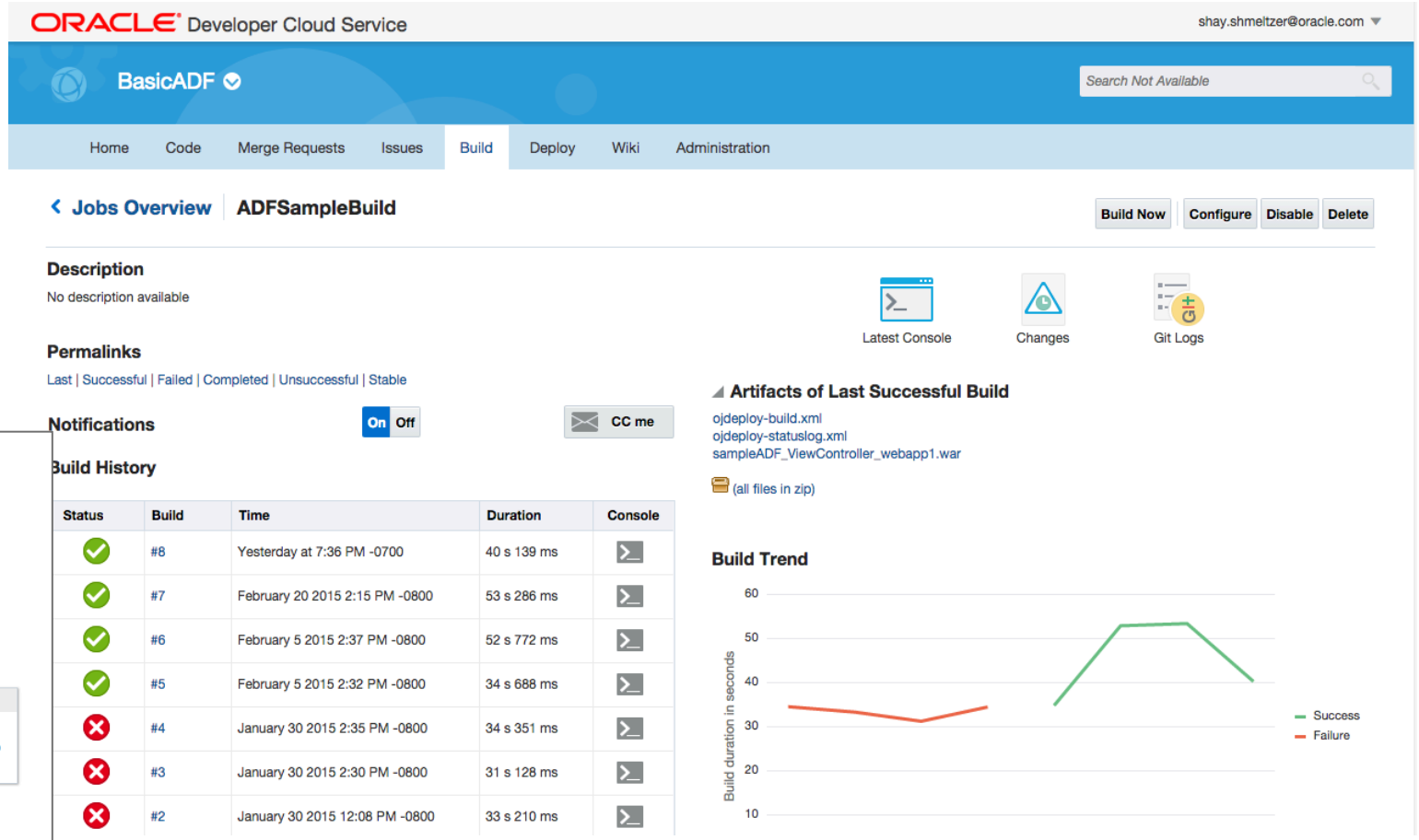
- Track Requirements/Bugs/ERs
- Assign to team members
- Integration with MyLyn in IDEs



Automate Project Builds

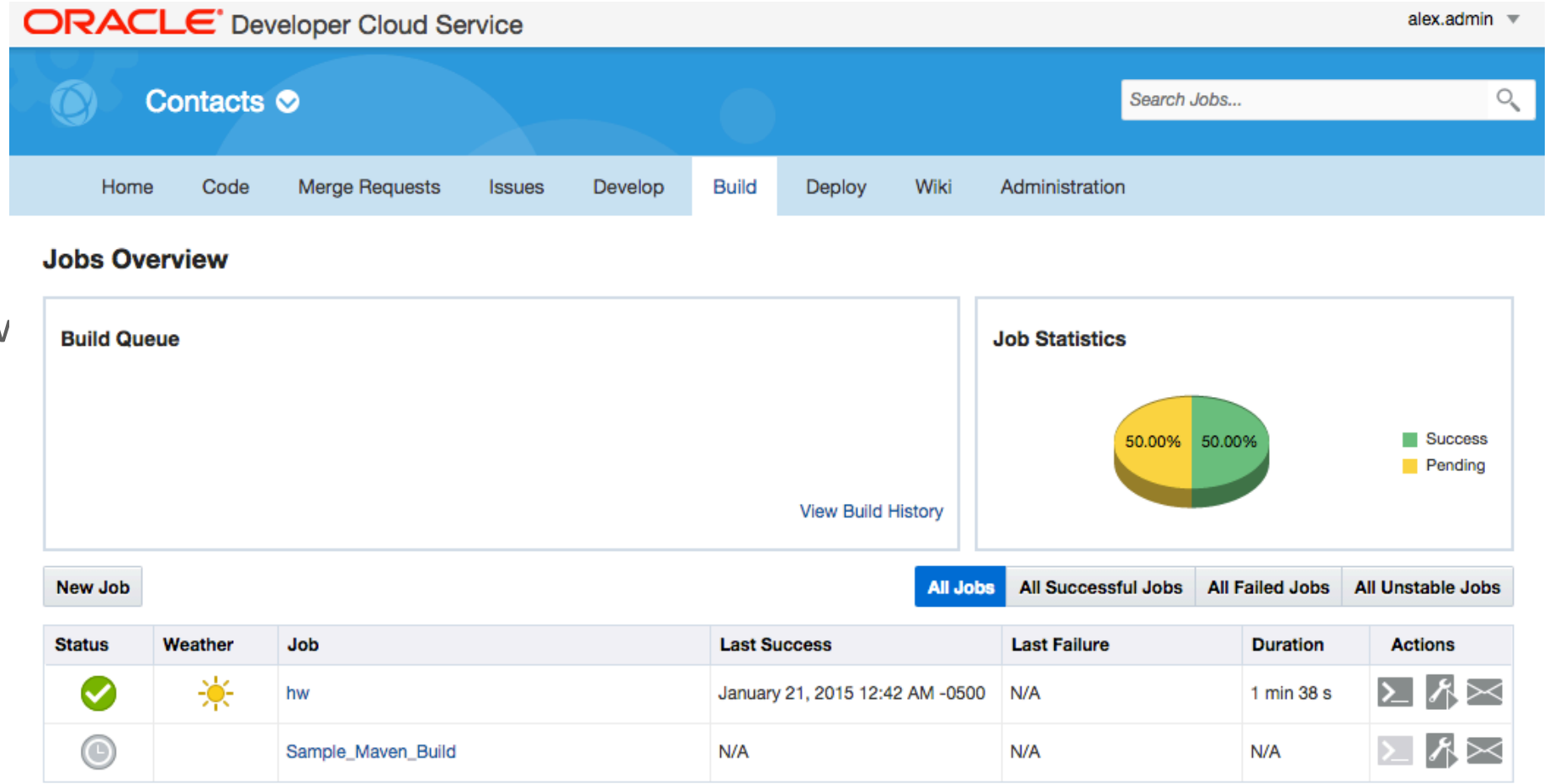
- Maven
- Ant
- Event based automation
  - Code check-in





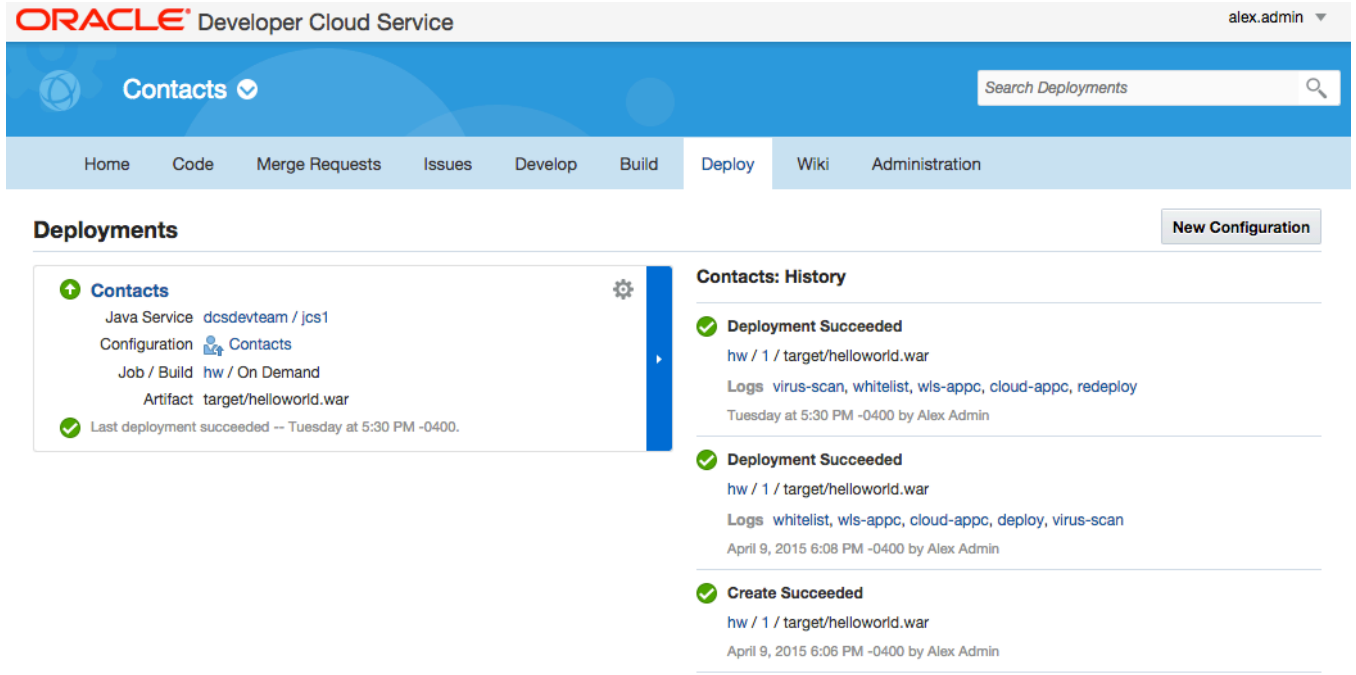
Continuous Integration / Continuous Delivery

- Build status by job
- Create new jobs
- View build history
- Save views
- Executor active view



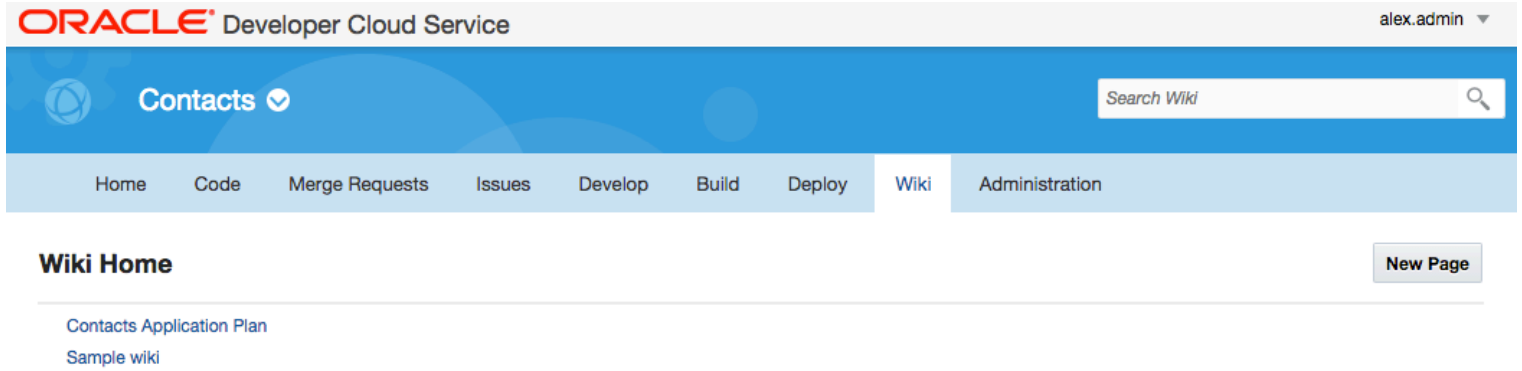
Simplified Application Deployment

- Create deployment configurations
- Start/Stop a deployment
- Redeploy/Un-deploy applications
- In the cloud or on-premise deployment



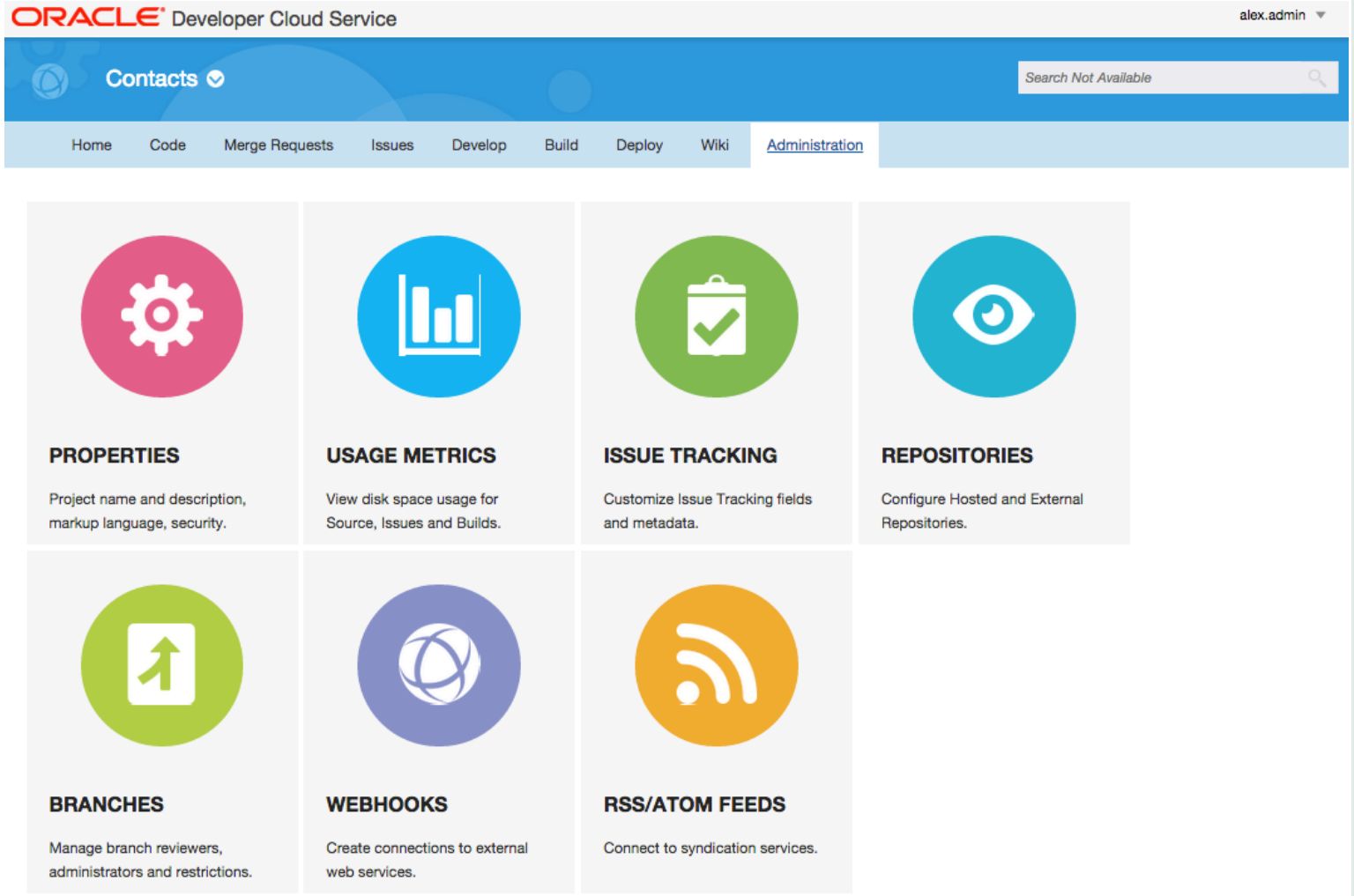
Share Information Through Wikis

- Create a new project wiki
- Collaboration through project specific wiki
- Content management
- Wiki markup of choice



Administer Your Projects

- Manage Project Properties
- Analyze Usage Data
- Customize Issue Tracking
- Configure Git Repositories
  - Hosted & External
- Manage Branches
  - Administrators & Restrictions
- Create Connections to External Web Services
- Connect to Syndication Services



Developer Cloud Service

Simplify Development

- Automated Provisioned Env
- Preconfigured & Integrated
- Automated Builds & Deployments
- Web based administration

Collaborate & Manage

- Integrated team source repository
- Continuous integration with breakage notifications
- Task/Defect tracking with activity stream and notifications

Deploy Automatically

- Deploy into Java Cloud Service automatically
- Workflow ensures build & test

Integrated With IDEs

- JDeveloper
- Eclipse
- NetBeans

Developer Cloud Service

DEMO

Source Control Management

Issue Tracking

Continuous Integration

Wiki Collaboration

Case Study

Team Overview – DevCS Development Team

- Distributed team of > 145 developers
- Thousands of lines of code
- Scrum methodology running 2 week sprints
  - Each sprint delivers production ready builds
- Uses Developer Cloud Service to manage all Development Operations (DevOps)

Source Management

- 27 Git repositories
- 1 Project for Developer Cloud Service
  - Represents many Git repositories for each product component and/or sub-component
  - Issue tracking spans project
- Logically separate code represents a component and has it's own Git repository
  - Versioned and branched independently

Merge Process – Merge Requests

- New features / bug fixes occur on a feature branch created by a developer
- Once feature / bug fixes are complete, merge request is created with target of master on component repository
- Default reviewers on a component repository feature branch
- Developer can add additional reviewers
- Code is reviewed
  - Iterated over based on reviewer comments -> Approved/Rejected
- Merge is initiated on Approved code
  - Every check-in that's merged initiates a build

Build Process

- Builds (In addition to check-in builds) – Development
  - 2 times a day an integration build is initiated
    - Builds all Git repositories for a consistent stripe in time
    - Build is deployed to VMs running in Cloud (Development Staging Env.)
    - Downstream job is initiated running functional test (selenium suite tests) against env.
- Builds – Production Candidates
  - Master is branched every 2 weeks and a Build is initiated
  - Deployed to a Production Candidate VM in Cloud
  - Selenium Tests run against Production Candidate VM in Cloud
  - Manual QA against Production Candidate VM
  - If all tests pass, branch may be deployed to customer production environment

Metrics Overview

- Many feature branch builds occur on every developer merge
- 2 integration builds/day
- 2 Week Development Sprint
  - 1 automated production release build every 2 weeks
- ~1,000 transactions/day
  - Transactions include commits, builds, code review activity, merges, tasks, etc…
- 145 Developers
- 27 Git repositories
- 1 Maven repository

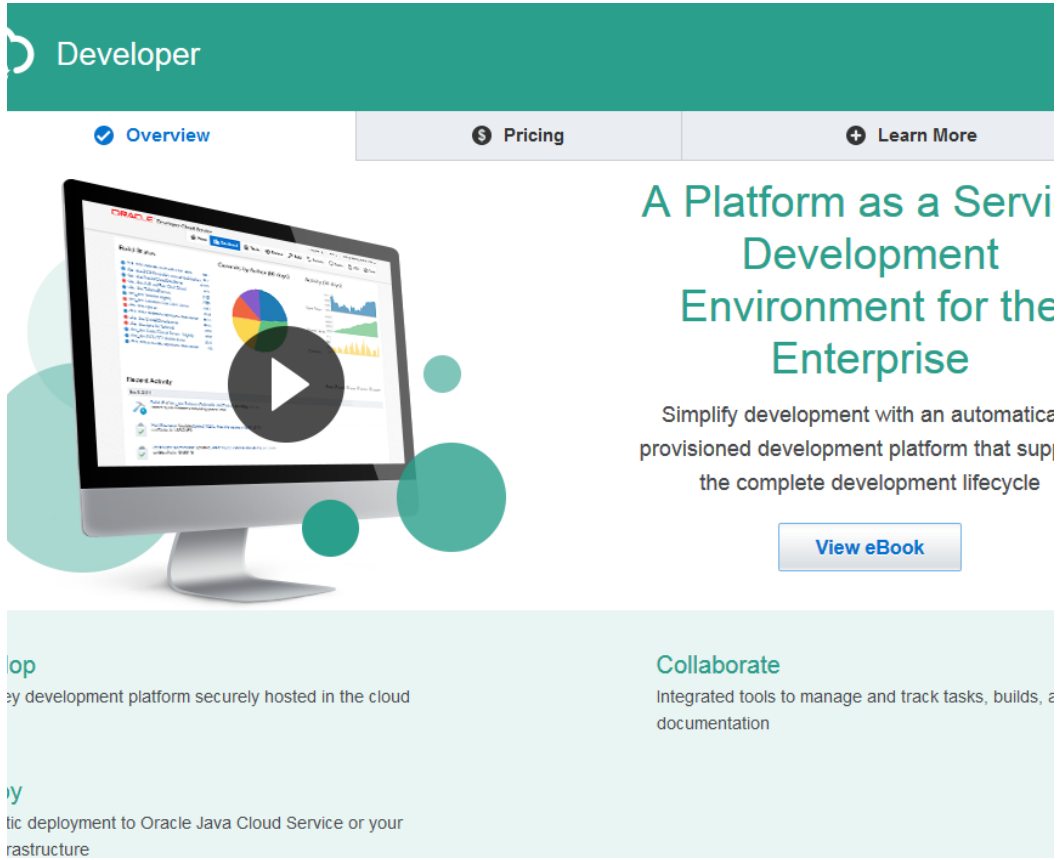
Summary

Developer Cloud Service Value Proposition

- Eliminate setup and startup time
- Reduce maintenance costs
- Leverage integrated ALM solution
- Extend code accessibility
- Improve team collaboration
- Simplify team management
- Streamline cloud deployment
- Produce better applications faster

Get Started Today

cloud.oracle.com/developer_service



- Tutorials
- Videos
- eBook
- Whitepapers
- Documentation
- Forums

Learn More

What	When	Where
HOL - Improved Development Lifecycle, Team Collaboration, and DevOps in the Cloud	Mon, 5:00	Hotel Nikko – Mendocino I/II
Oracle Cloud Platform for Rapid Applications Development and Integration in the Cloud	Tue, 12:15	Moscone South 302
Development Operations in the Cloud: A Use Case and Best Practices	Tue, 5:30	Parc 55 – Powell I/II
HOL - Improved Development Lifecycle, Team Collaboration, and DevOps in the Cloud	Wed, 2:45	Hotel Nikko – Mendocino I/II
DevOps for Mobile in the Cloud	Thu, 12:00	Moscone South 304

JavaOne™

ORACLE®