



Harnessing DevOps for Streamlining Integration Delivery

| Peter Antony, Senior Solutions Principal
Technical Advisory Services, June 2024

boomi

— Agenda

01 Introduction

02 The Why

03 CI/CD Platforms

04 Implementation & Workflow

05 GitHub Actions Demo

Introduction



Introduction

What is DevOps and CI/CD?

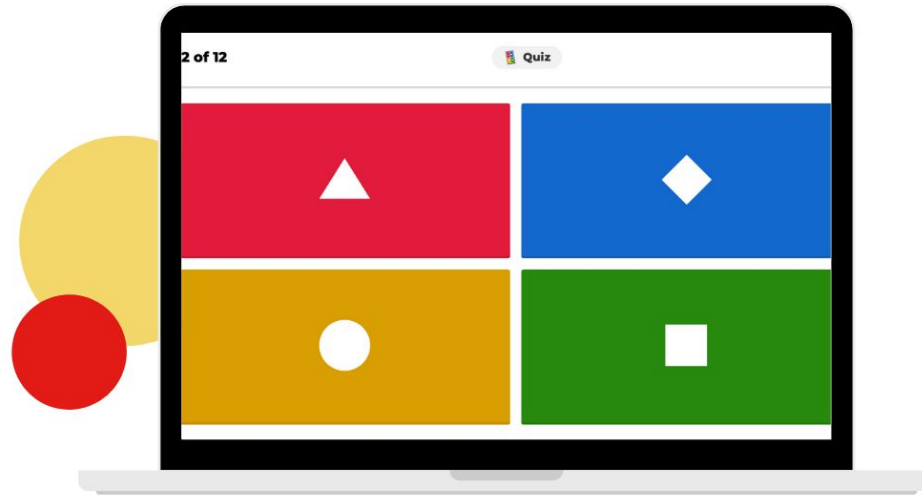
- **DevOps:** Approach that bridges the gap between software development and IT operations
 - Goal is to streamline software delivery through automation
 - Emphasis on collaboration and communication
- **CI/CD:** Continuous Integration/Continuous Deployment, is a software development practice that allows code changes to be tested and deployed quicker and reliably
 - **CI:** Automated integration of code changes into a shared repository
 - **CD:** Automated deployment of code changes to environments
- **Key Components:** Automation, Collaboration, Continuous Monitoring, Governance

Introduction

What about Boomi and CI/CD?

- CI/CD is implemented using a toolchain
- Boomi (AtomSphere API) is integral to CI/CD
- Enables the delivery of software (Boomi Packages)
- Benefits are increased as toolchain is matured and governed
 - Shorter release cycles that deliver better quality Components
 - Reduces cost
 - Minimising multiple handoffs/approvals
 - Reusability in automation

CI/CD In Your Organisation



Kahoot!

CI/CD Components

Sample Toolchain



Boomi AtomSphere

iPaaS used to connect multiple segments of an organisation's digital ecosystem. Features low-code environment, scalable to suit organisation needs and centralised platform.



CI/CD Orchestration

Continuous Integration, Delivery, and Deployment tool available as part of GitLab. Used to orchestrate pipelines of jobs that perform specific tasks.



Work Management

Help teams to effectively work through better collaboration, tracking and organisation of tasks.



Code Quality Analysis

Inspect processes to detect violations of custom rules. Promote better development through better practices.



Testing

Assists with the Automated Testing of Boomi integrate processes. Includes Functional and Performance testing capabilities.



API Security Testing

Help identify and assess vulnerabilities to protect against unauthorised access, data breaches, attacks and other risks.



Sample Package to Deploy and Test

Composition

Multiple components are packaged together in a bundle and deployed to different Environments

The screenshot displays the Boomi API Manager interface. On the left, a tree view shows the project structure under 'Work - CI/CD+DevOps - Dummy API', including folders for APIs, Connectors, Cross Reference Tables, Process Properties, and Processes. The 'Processes' folder is expanded, showing 'Sub: delegate', 'Unit Test: Dummy', 'Unit Test: Dummy 2', 'WSS: Dummy Fixed Reply', and 'WSS: Dummy Request Reply'.

The main panel shows the 'REST Configuration' for the 'Dummy API - API Service'. The 'REST' tab is active, displaying the 'Path to REST' as '/ws/rest/dummy'. Below this, two endpoint configurations are listed:

Actions	Method	Resource Path	Process	Endpoint Description	Headers
	GET	—	WSS: Dummy Fixed Reply	—	—

[Delete the Dummy object and all its endpoints](#)

Actions	Method	Resource Path	Process	Endpoint Description	Headers
	GET	—	WSS: Dummy Request Reply	—	—

[Delete the DummyRequest object and all its endpoints](#)

At the bottom, the 'WSS: Dummy Request Reply - Process' configuration is shown. It includes a search bar and a toolbar with options like 'Map', 'Set Properties', and 'Message'. The process flow diagram consists of the following steps: 'Start - Web Services Server Dummy Request Get', 'DPP_log_and_benchmark', 'Sub: delegate', 'Set Properties', 'log parameter?', a decision diamond 'log parameter?' with 'True' and 'False' paths, 'Log Payload', and 'Return Documents'. Buttons for 'Create API' and 'Create Packaged Component' are visible in the top right.

The Why



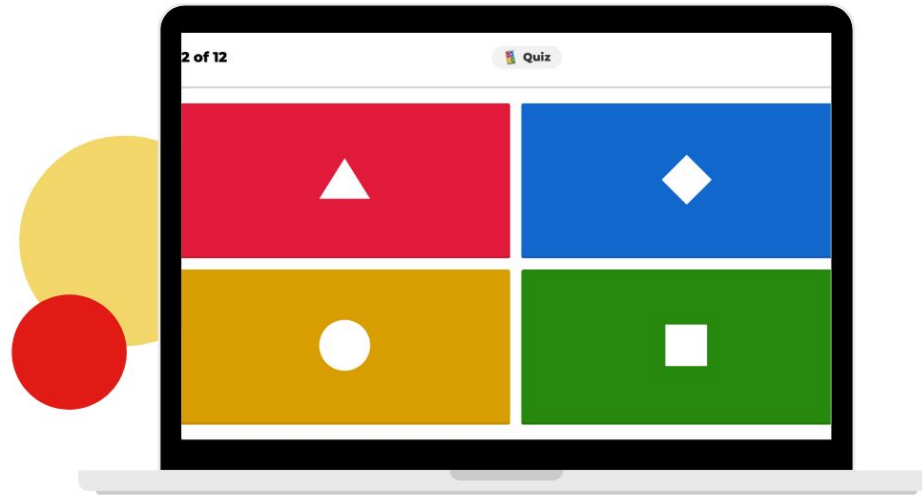


Achieving peace of mind?

Challenges

- Use of Manual processes
- How to reduce Release cycle times while improving quality
- Reducing costs of errors and re-work
- End-to-end monitoring and visibility
- How to minimise multiple handoffs/approvals

Tell us about your DevOps Team



Kahoot!

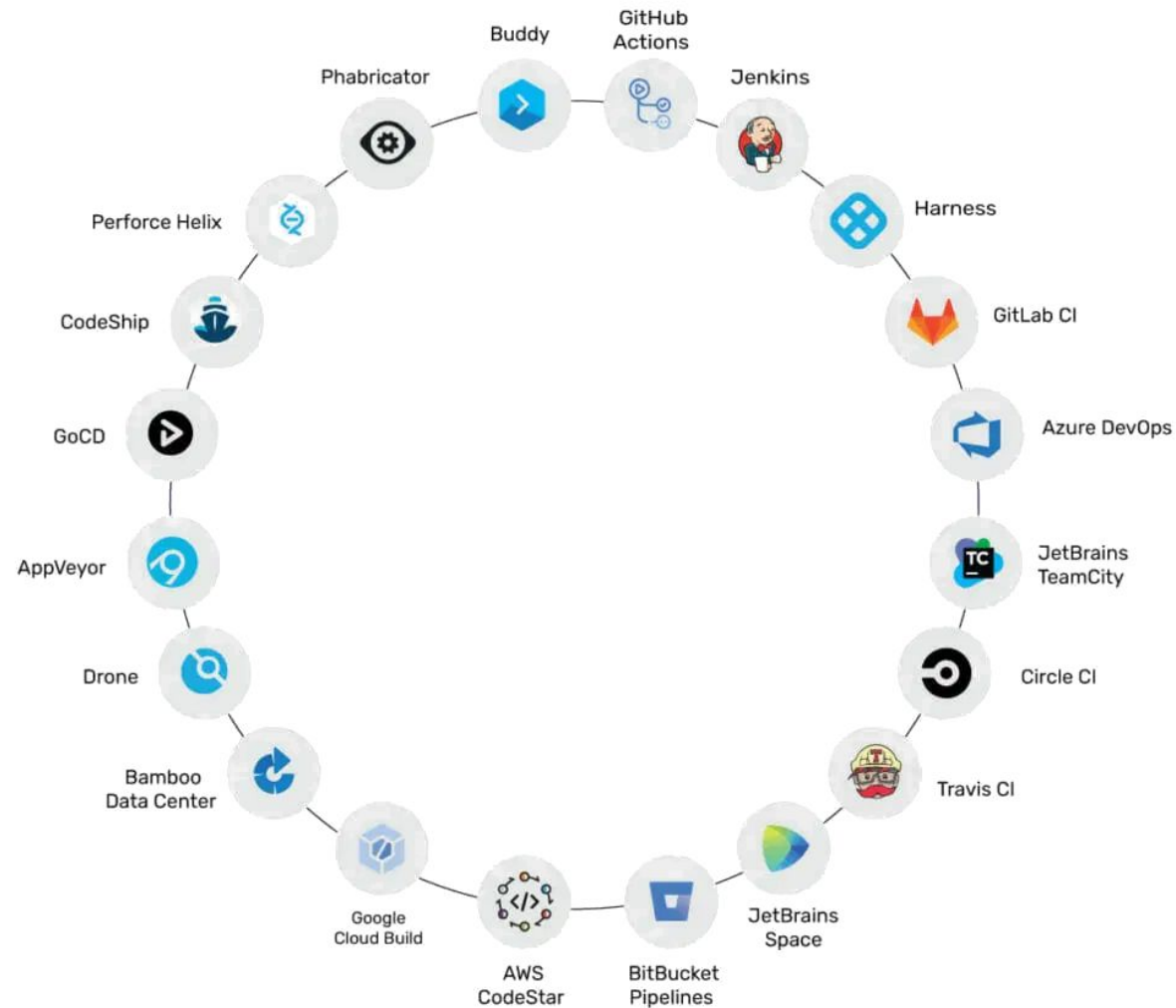
Benefits

- Benefits are increased as toolchain is matured and governed
- Time to delivery is reduced through automation
- Reduced costs (identification through early feedback)
- Removal of manual checklists
- Reusability and consistency through Governance and automation

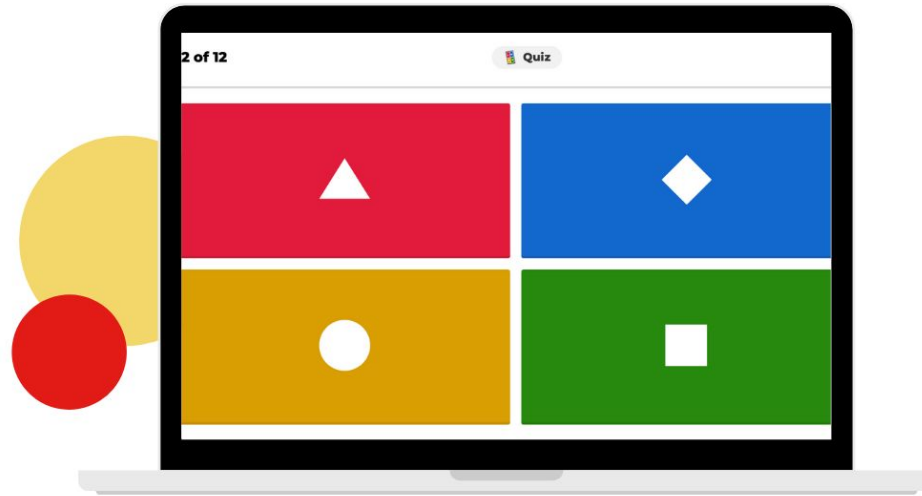
CI/CD Platforms



(Some) CI/CD Platforms



Ok, time to spill the beans



Kahoot!

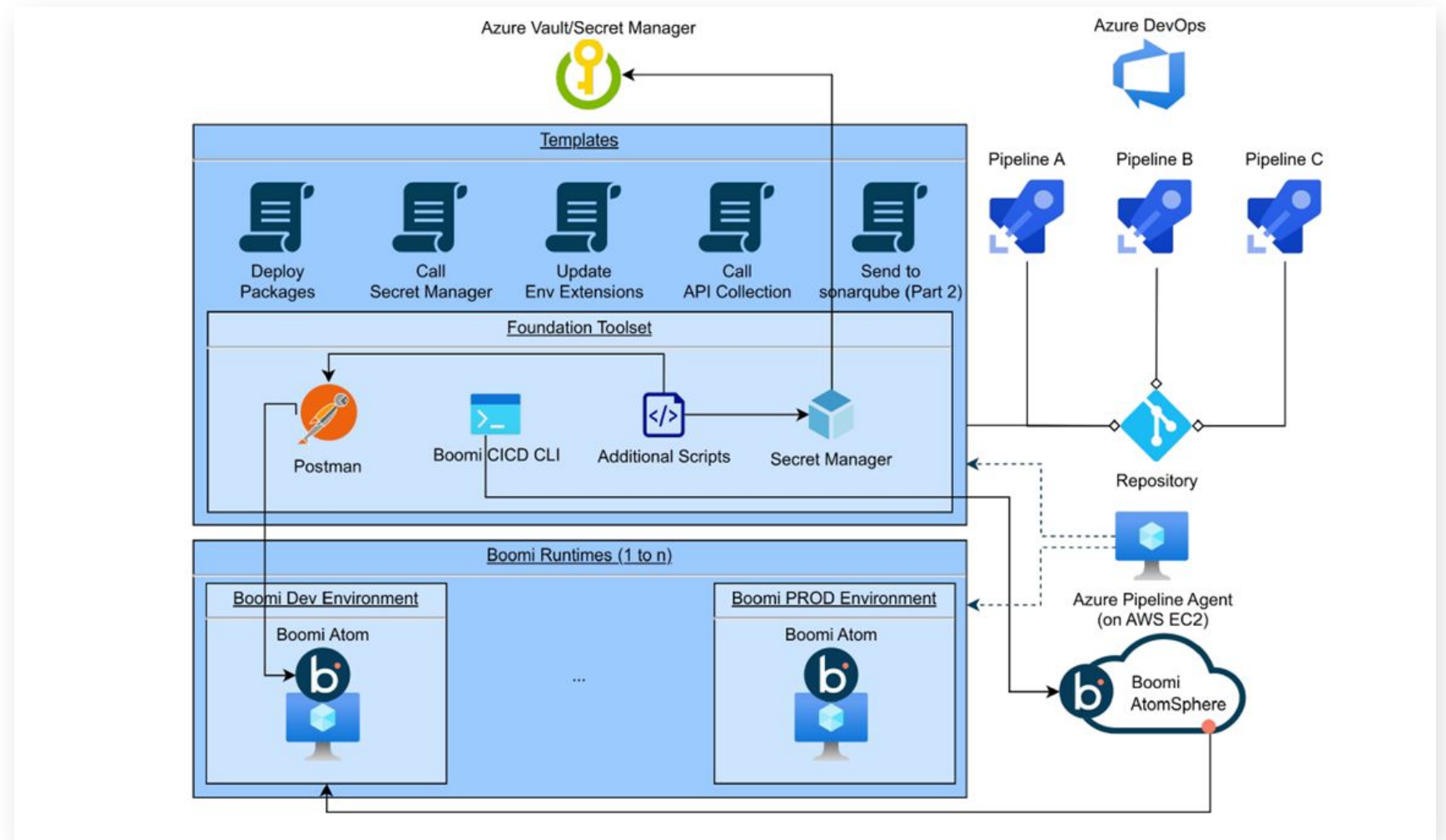
Common CI/CD Functionality

- **Continuous Integration**
 - Frequent Code integration
 - Build Triggers
- **Continuous Deployment**
 - Deployment Automation
 - Deployment Triggers
- **Pipeline Orchestration**
 - Multi-step workflows and parallel execution
 - Governance, compliance and monitoring

Azure DevOps

Toolchain Workflow

Multiple components are working together to deliver the required functionality



Azure DevOps

Pipeline Definition

Components are configured that provide information on what to deploy and what functions to perform

```
parameters:
  - name: packageVersion
    displayName: Package Version, leave 0 for a new deployment OR put the Build Number to reuse an existing package.
    type: string
    default: 0
  - name: notes
    displayName: Notes
    type: string
    default: 'Dummy API Bundle'
  - name: deployToProd
    type: boolean
    default: false

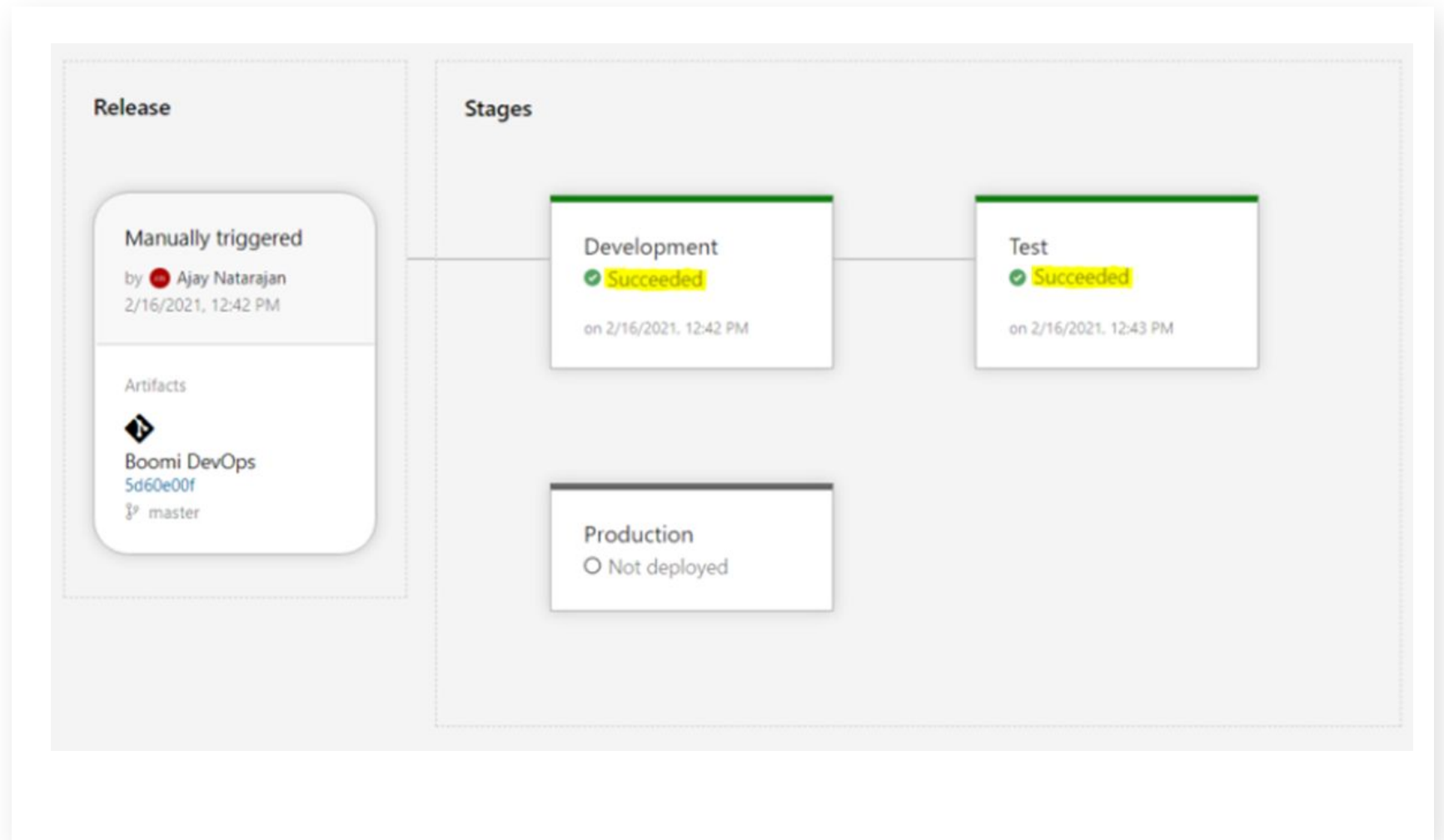
trigger:
  - master

pool:
  name: Default
  demands:
    - agent.os -equals Linux
```

Azure DevOps

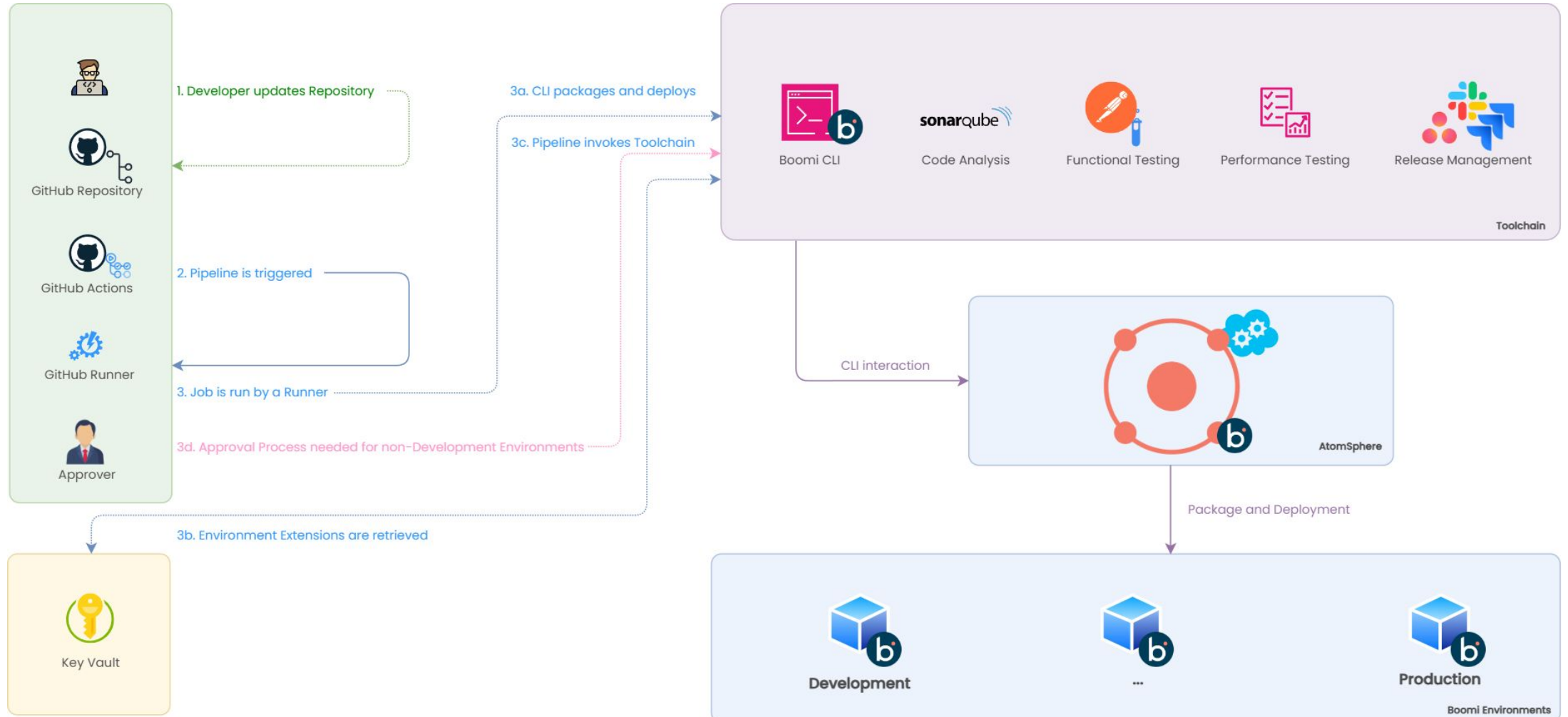
Pipeline Monitoring

Realtime feedback
provided on individual
jobs



GitHub Actions

Toolchain Workflow



GitHub Actions

Pipeline Definition

Different approaches can be taken to Pipeline configuration including a single file per Environment or a single file that caters to multiple Environments.

It needs to include the necessary steps to run the Toolchain components

```
name: Build and Deploy Package
on:
  push:
    branches:
      - main
    paths:
      - 'InterfaceDefs/*.mf.Test'

env:
  AWS_ACCESS_KEY_ID: ${ secrets.ID }
  AWS_SECRET_ACCESS_KEY: ${ secrets.ID2 }
  AWS_DEFAULT_REGION: ap-southeast-2
  AWS_DEFAULT_OUTPUT: json

  ID: ${ github.run_id }-${ github.run_attempt }

jobs:
  preReqs:
    runs-on: [self-hosted, linux, X64]
```

GitHub Actions

Interface Manifest

A single Manifest file is created for a Package bundle that defines the Components and what to do.

This can include Environment Extension updates, Testing and Ticket management

```
##Build
componentId=12345678-abcd-1111-2222-123456789012,87654321-dcba-2222-1111-210987654321
componentType=process
ver=0.5.181
buildNotes=Packaged from GitHub

##Deployment
#deploy=1.Dev
deploy=2.Test
#deploy=3.Prod
listenerStatus=RUNNING
deployNotes=Deployment from GitHub

##Environment Extensions
devEnvUpd=0
devEnvId=12345678-abcd-1111-2222-123456789012
devEnvCompType=connection
```

GitHub Actions

Environment Approvals

Approvals can be configured as part of Pipeline execution if needed.

This shows that approval is needed for any deployment to a non-Prod environment

Environments

New environment

You can configure environments with protection rules, variables and secrets. [Learn more about configuring environments.](#)

Prod	1 protection rule		
Test	1 protection rule	1 variable	
Dev			

GitHub Actions

Code Quality Analysis

Components can be examined to ensure Development rules are followed.

SonarQube or operating system tools can be used to navigate and check process definitions for issues

☆ boomicid Passed Last analysis: 34 minutes ago

🐛 Bugs 1 C | 🔒 Vulnerabilities 0 A | 🛡️ Hotspots Reviewed — A | 🗑️ Code Smells 2 A | Coverage — | Duplications 0.0% ○ | Lines 531 XS XML

List of Packaged Components Code Quality Report

Name	Version	Type	Issue
[wsvc] IP Address	154	process	Process description must be set

GitHub Actions

Environment Extensions

Sensitive Environment Extension values can be stored in a Vault for inclusion during Pipeline runtime

Secret details

Encryption key

aws/secretsmanager

Secret name

boomi/1.Test-AtomCloud-US

Secret ARN

arn:aws:secretsmanager:ap-southeast-1:622467680580:secret:boomi/1.Test-AtomCloud-US-bSNnNb

Secret description

Endpoint URL, user and password used to access Boomi Web Services Services on Integration (Test Atom Cloud)

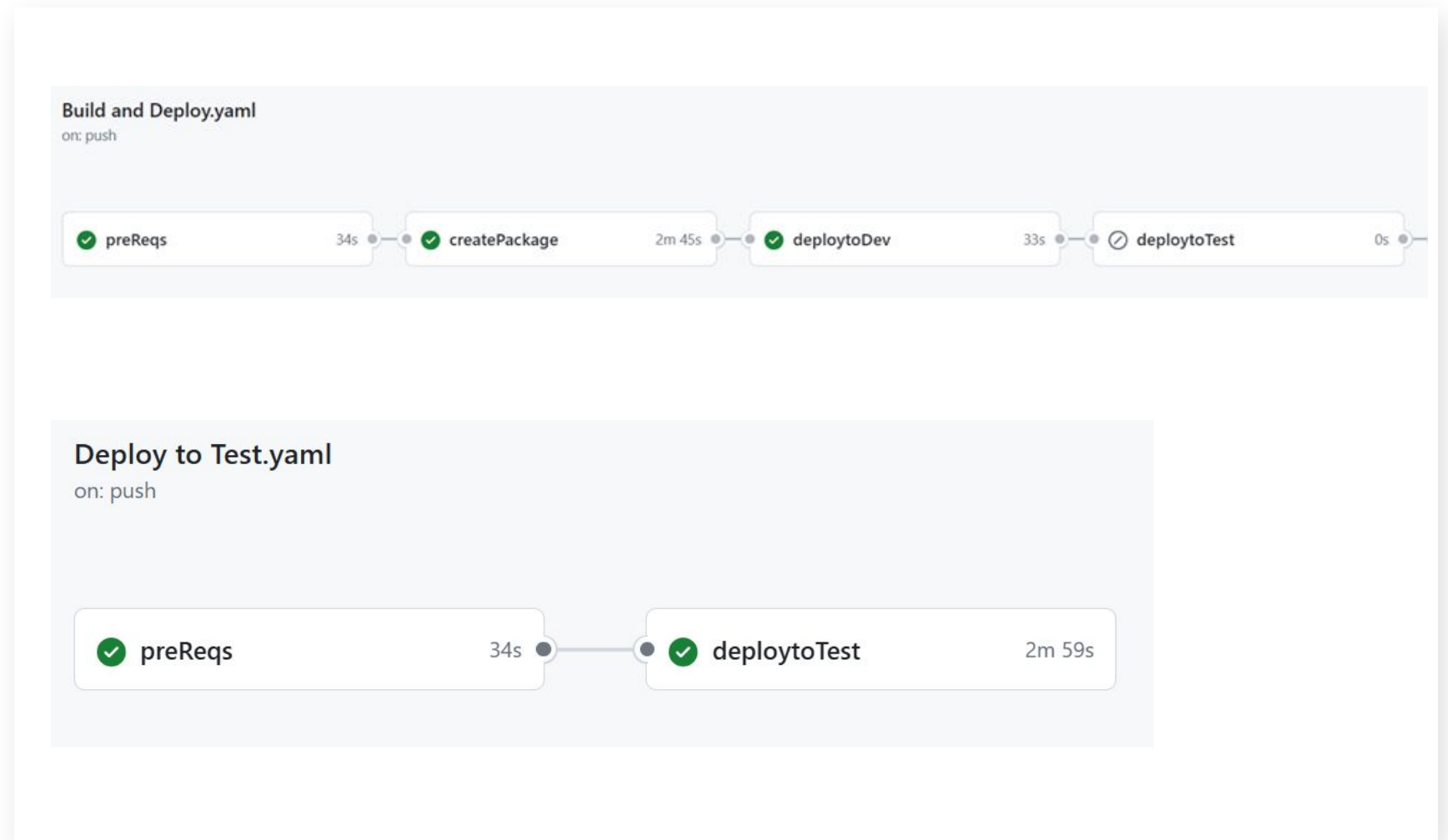
Using in a Pipeline

```
aws secretsmanager get-secret-value --secret-id MyTestSecret
```

GitHub Actions

Pipeline Monitoring

Approvals and appropriate Pipeline configuration provides the necessary Governance controls



GitLab CI/CD

Pipeline Definition

Components are configured that provide information on what to deploy and what functions to perform

```
stages:
  - build
  - deploy

build-job:
  stage: build
  except:
    changes:
      - ".gitlab-ci.yml"
      - "config.toml"
      - "*.env"

script:
  - echo $BOOMI_USERNAME
  - _BOOMI_PROC=$(git diff --name-only $CI_COMMIT_BEFORE_SHA $CI_COMMIT_SHA)
  - _BOOMI_COMP_ID=$(grep ^compId= "$_BOOMI_PROC" | cut -f2 -d'=')
  - _BOOMI_VER=$(grep ^version= "$_BOOMI_PROC" | cut -f2 -d'=')
  - _BOOMI_NOTES=$(grep ^notes= "$_BOOMI_PROC" | cut -f2 -d'=')
  - /home/peter/src/Boomi/initcrPackage.sh authToken=$BOOMI_USERNAME componentId=$_BOOMI_COMP_ID
```

GitLab CI/CD

Pipeline Context

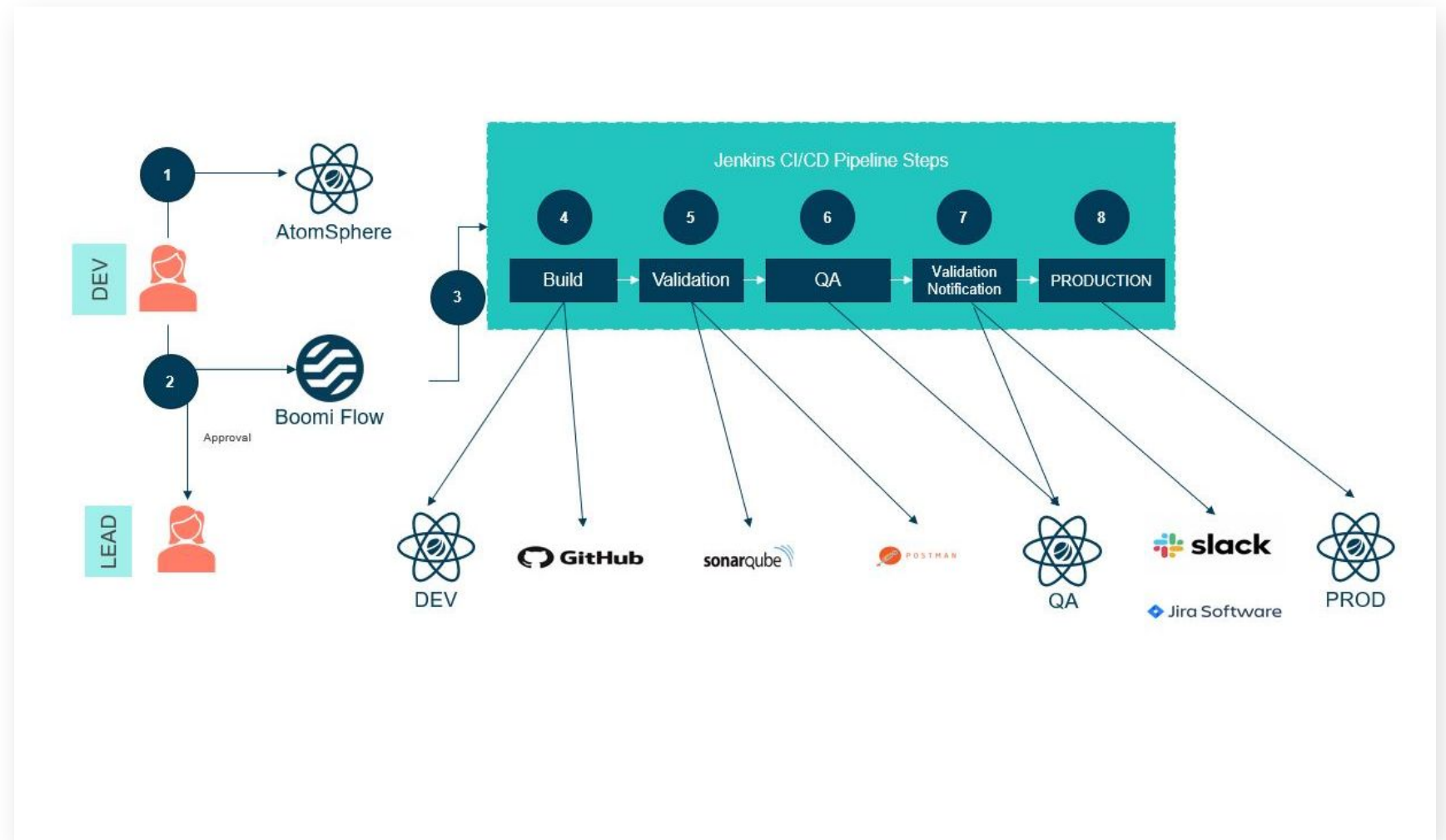
Boomi components are provided as input to the pipeline execution

```
compId=5d40981f-bc24-47c2-92a0-86ce7832f2e0  
version=0.6.2  
notes=Packaged from GitLab  
targetEnv=test
```

Jenkins

Toolchain Workflow

Multiple components work together to deliver required functionality



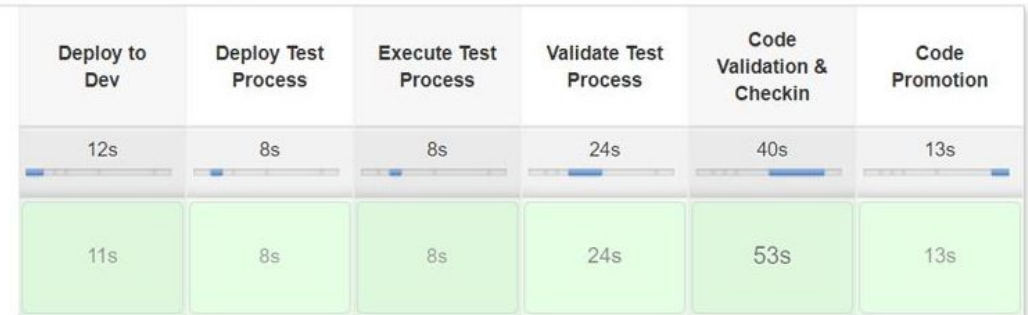
Jenkins

Jobs

Discrete functionality delivers the toolchain capabilities

Package Deployment	Process Deployment	Installation	Operations
<ol style="list-style-type: none">1. Package Creation2. Package Deployment3. Test Execution4. Test Verification5. Sonar Validation6. GIT check-in7. Orchestration	<ol style="list-style-type: none">1. Deploy Process(es)2. Promote Process(es)3. Test Execution4. Test Verification5. Sonar Validation6. GIT check-in7. Orchestration	<ol style="list-style-type: none">1. Create Cloud Atom2. Create Local Atom3. Create Environment, Roles4. Get Installation Token5. (De)Attach Atoms	<ol style="list-style-type: none">1. Process Execution2. Record Status3. Toggle Listener Status4. Toggle Schedule Status5. Schedule Management

Continuous Integration & Deployment Pipeline



CI/CD Components

Sample Toolchain



Boomi AtomSphere

iPaaS used to connect multiple segments of an organisation's digital ecosystem. Features low-code environment, scalable to suit organisation needs and centralised platform.



CI/CD Orchestration

Continuous Integration, Delivery, and Deployment tool available as part of GitLab. Used to orchestrate pipelines of jobs that perform specific tasks.



Work Management

Help teams to effectively work through better collaboration, tracking and organisation of tasks.



Code Quality Analysis

Inspect processes to detect violations of custom rules. Promote better development through better practices.



Testing

Assists with the Automated Testing of Boomi integrate processes. Includes Functional and Performance testing capabilities.



API Security Testing

Help identify and assess vulnerabilities to protect against unauthorised access, data breaches, attacks and other risks.



CI/CD Components

AtomSphere API at the Core



Boomi AtomSphere

iPaaS used to connect multiple segments of an organisation's digital ecosystem. Features low-code environment, scalable to suit organisation needs and centralised platform.



CI/CD Orchestration

Continuous Integration, Delivery, and Deployment tool available as part of GitLab. Used to orchestrate pipelines of jobs that perform specific tasks.



Work Management

Help teams to effectively work through better collaboration, tracking and organisation of tasks.



Code Quality Analysis

Inspect processes to detect violations of custom rules. Promote better development through better practices.



Testing

Assists with the Automated Testing of Boomi integrate processes. Includes Functional and Performance testing capabilities.



API Security Testing

Help identify and assess vulnerabilities to protect against unauthorised access, data breaches, attacks and other risks.



Implementation & Workflow





Path to Growth

CI/CD Components

Sample Toolchain



Boomi AtomSphere

iPaaS used to connect multiple segments of an organisation's digital ecosystem. Features low-code environment, scalable to suit organisation needs and centralised platform.



CI/CD Orchestration

Continuous Integration, Delivery, and Deployment tool available as part of GitLab. Used to orchestrate pipelines of jobs that perform specific tasks.



Work Management

Help teams to effectively work through better collaboration, tracking and organisation of tasks.

Code Quality Analysis

Inspect processes to detect violations of custom rules. Promote better development through better practices.



Testing

Assists with the Automated Testing of Boomi integrate processes. Includes Functional and Performance testing capabilities.



API Security Testing

Help identify and assess vulnerabilities to protect against unauthorised access, data breaches, attacks and other risks.



CI/CD Components

Sample Toolchain



Boomi AtomSphere

iPaaS used to connect multiple segments of an organisation's digital ecosystem. Features low-code environment, scalable to suit organisation needs and centralised platform.



CI/CD Orchestration

Continuous Integration, Delivery, and Deployment tool available as part of GitLab. Used to orchestrate pipelines of jobs that perform specific tasks.



Work Management

Help teams to effectively work through better collaboration, tracking and organisation of tasks.

Code Quality Analysis

Inspect processes to detect violations of custom rules. Promote better development through better practices.



Testing

Assists with the Automated Testing of Boomi integrate processes. Includes Functional and Performance testing capabilities.



API Security Testing

Help identify and assess vulnerabilities to protect against unauthorised access, data breaches, attacks and other risks.





demo.

GitHub Actions



Boomi Professional Services

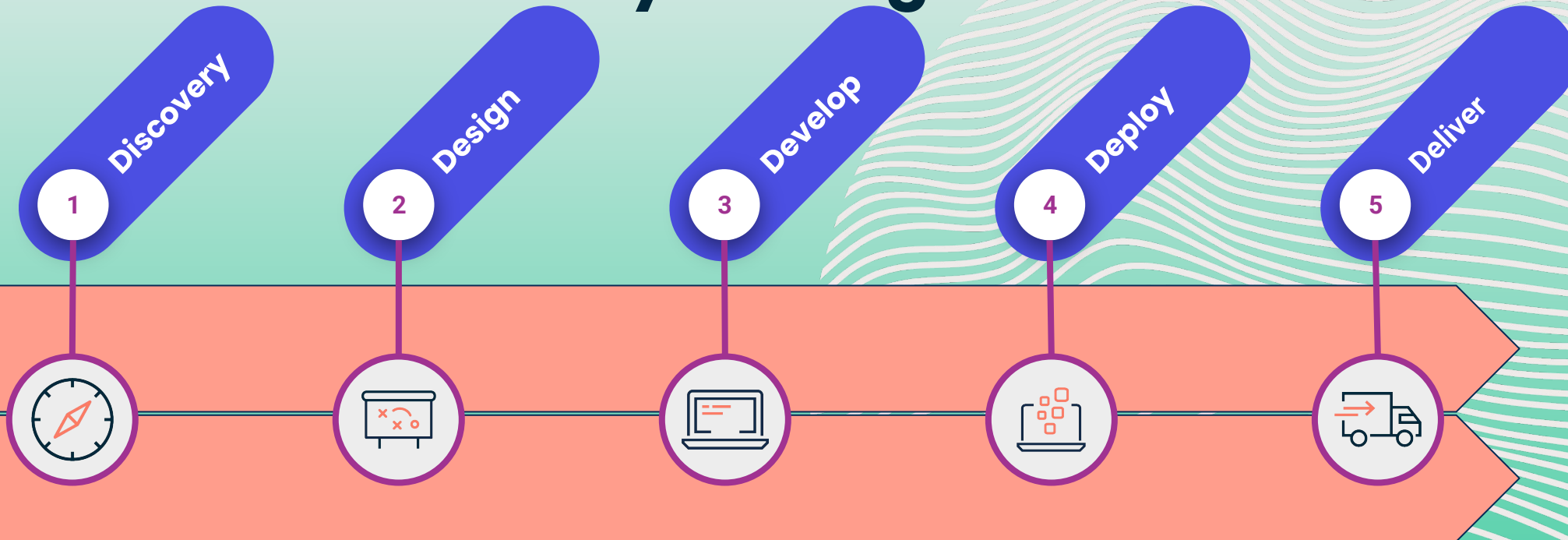
Gourav Agrawal

Enterprise Solutions Manager

Boomi PSO

boomi

We Support You All the Way Through



Boomi Ecosystem

Boomi Community

Global Network of Peer Consultants

Customer Success Manager

Boomi Product Management

Platform Support
Standard, Premier, and Premier+

Boomi Professional Services Organization

Professional Services

- **Consulting**
Engage Boomi certified consultants to deliver complete production-ready solutions or to augment your development teams.
- **Technical Architecture**
Collaborate with Boomi's architectural services to ensure your Boomi solution design and environment aligns with recommended practices.
- **Project Management**
Apply our leadership in project methodologies to ensure a successful delivery meeting quality expectations, on-time, and within budget.

Managed Services

- **Managed Cloud Services**
Ready-built, tried, tested, secure, proven hosted cloud platform optimized for Boomi Atmosphere.
- **Insights**
Provides a holistic view of key metrics across your entire Boomi Integration landscape, so you can easily monitor, troubleshoot, and optimize all your Boomi operations.
- **Process Management**
Comprehensive monitoring and management for the "run" phase of a Boomi implementation.

Technical Advisory Services

- **Enterprise Architecture**
Seasoned Boomi Platform Architect and Enterprise Architect to provide expert guidance on developing new strategies and IT capabilities based on the Boomi Platform.
- **Technical Account Manager**
A trusted Boomi advocate provides ongoing, proactive technical engagement and incident monitoring

Delivery Assurance Program (DAP)

The Boomi Delivery Assurance Program ensures your organization receives the appropriate architecture and process guidance at critical stages of your Boomi Platform implementation to avoid pre- or post-deployment issues while allowing for scalability in the future.

- Implement based on recommended practices
- Leverage Boomi's elite team of integration professionals
- Develop skills in-house
- Learn ways to maximize your technology investment

Architecture Review Checkpoint

Review and advise your team on:

- Integration scope
- Runtime analysis
- Governance
- Error handling
- Account management
- Integration maintenance

Process Review Checkpoint

Review and provide assessment for:

- Process performance
- Process scalability
- Usability/maintenance
- Testing
- Adherence to best practices

Discovery

Design

Develop

Deploy

Deliver

Resource Team



Technical Architect

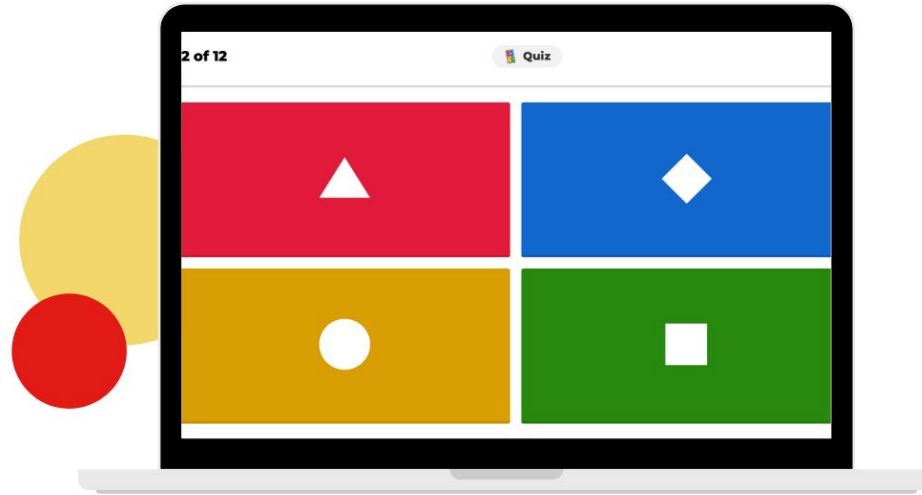


Technical Consultant



Project Manager

Please provide your Feedback



Kahoot!



Boomi World Tour Sydney 21st November 2024

Save the date coming soon!





We are Boomi.

Be Bold. Be You. Be Boomi.