



EUROPE'S LEADING AEM DEVELOPER CONFERENCE
25th – 27th SEPTEMBER 2023

Automated CI with AEM Cloud service

Andres Bott, Jakub Wądołowski

Automated CI with AEM Cloud service

Agenda:

- Intro
- About CI/CD systems and AEMCS
- How to use RDEs for
Continuous Integration
- Customer perspective



Andres Bott
Adobe



Jakub Wądołowski
Diva-e

How did we get here

Getting in touch with Diva-e

Early 2022 Jakub and I got connected

Conversation on different subjects related to testing

Pain point:

- Cloud Manager feedback on prod deployments is too late.

Work on POC how Diva-e could run tests close to CM.

RDEs became a thing around the same time

From developer to production

From developer to production

Developer tests locally (SDK)

- *mvn clean verify*
- Manual testing

A large white arrow pointing to the right, containing the text 'Local testing'. Below the arrow is a white icon of a computer monitor.

Local testing



From developer to production

Developer tests locally (SDK)

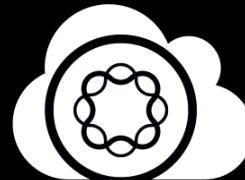
Functional tests on deployments

- Code is already merged and pushed to Cloud Manager

Local testing



System testing

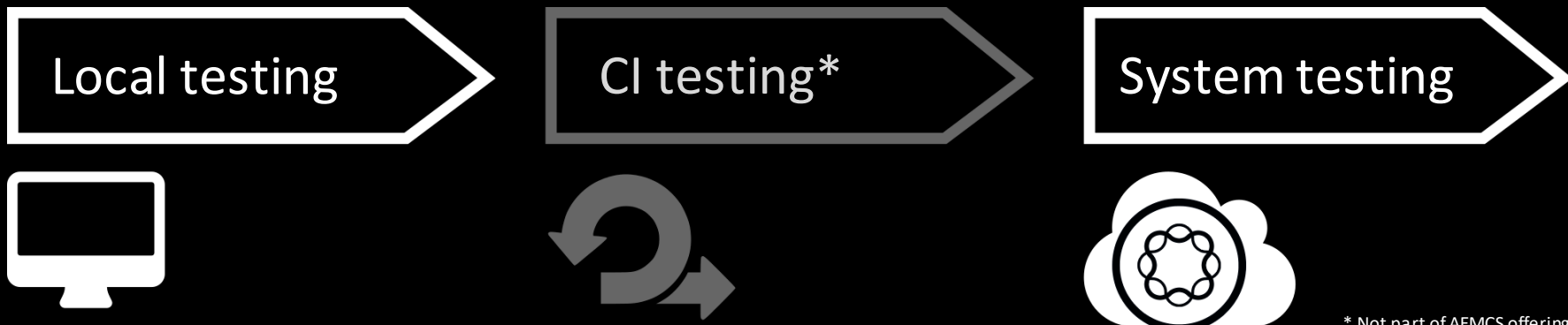


From developer to production

Developer tests locally

Functional tests on deployments

Missing part: *quick code iterations on branches or PRs*



* Not part of AEMCS offering

When projects grow:

Build step in CI

+ Functional tests / Integration tests

1. Deploy code on an environment.
2. Blackbox automated verification.
 - Mostly doing Http requests to the system.

Where do we deploy our changes?

Deployment Targets

Do nothing:

- Accept the risk of less verification.
- Slower feedback loop.
- Harder to debug issues.

Deployment Targets

Do nothing

Replicate cloud service with SDK(s):

- Start author + publish SDK and test against them. (e.g. on a VM)
- (alternative) Containerize SDKs, deploy on own K8S, etc.
- System is not comparable to CS: Mongo, replication, etc.
- Expensive maintenance work.

Deployment Targets

Do nothing

Replicate cloud service with SDK(s)

Use AEMCS Dev/sandbox environments:

- Pipelines are slow (30-60 min)
- API limitations
- Code needs to be pushed to CM

Deployment Targets

Do nothing

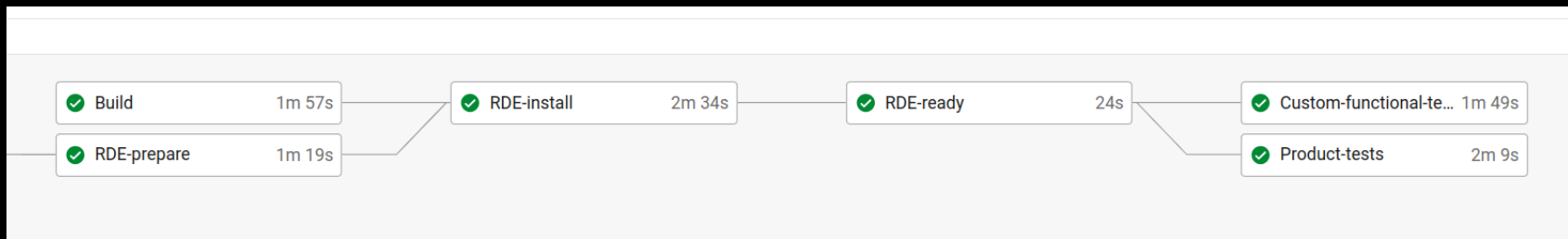
Replicate cloud service with SDK(s)

Use AEMCS Dev/sandbox environments

RDEs to the rescue: 

- Fast deployments (3-4min)
- Isolated reproduceable environment
- Very similar to production

CI with RDEs



- Sample using CircleCi
- Other systems are easy as well

Sample:

<https://github.com/andresbott/aem-guides-wknd/blob/main/.circleci/config.yml>

Start from a clean state:

```
aio aem:rde:reset
```

RDE reset at the start

- Usually take ~10-15 min
- At the end of the pipeline do a "fire and forget" reset
- We can check if RDE is already clean

Tests require an "admin" like user:

- Needed to interact with AEM features
- Options:
 - Content package
 - OSGI config with repointit

Tests require an "admin" like user:

- OSGI config with repointit

```
aio aem:rde:install -t osgi-config \
```

```
org.apache.sling.jcr.repointit.RepositoryInitializer~testadmin-user-add.cfg.json
```

```
delete user test-admin
```

```
create user test-admin with password <pass>
```

```
add test-admin to group administrators
```

- Install project:

```
aio aem:rde:install -t content-package <package.zip>
```

```
aio aem:rde:install -t dispatcher-config <dispatcher-config.zip>
```

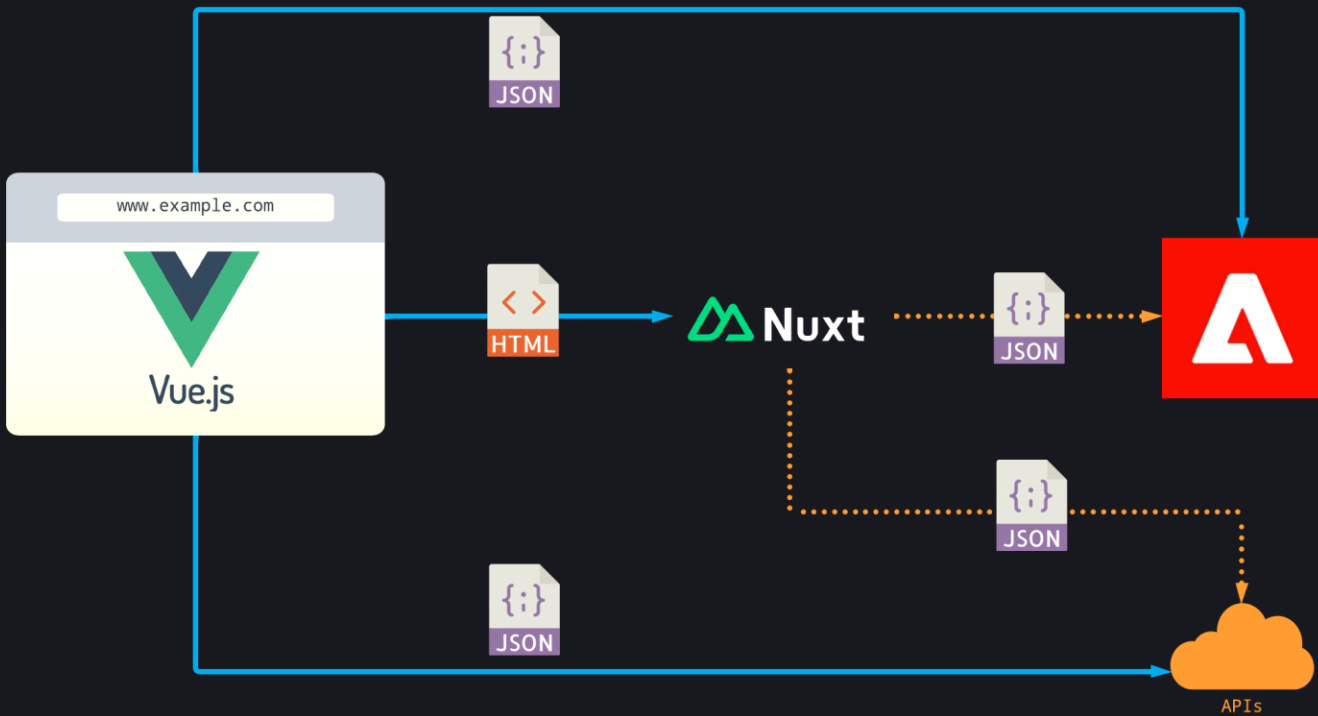
- AEM might restart sling bundles,
 - the CI system needs to handle the restart.
 - wait until publish returns status code 200
 - wait until publish page contains <title>= "..."

Let's look into the tests

- Java ITs: Custom functional
- Product tests (Exploratory using Adobe internal platform)
- Lighthouse performance validation
- Others:
 - UI tests, cypress, selenium
 - ... ideas ?

From the Customer perspective

Platform overview



Setup complexity

- E-commerce platform
- Dozens of services
- AEMaaCS exposes headless content API
- Interconnections & dependencies

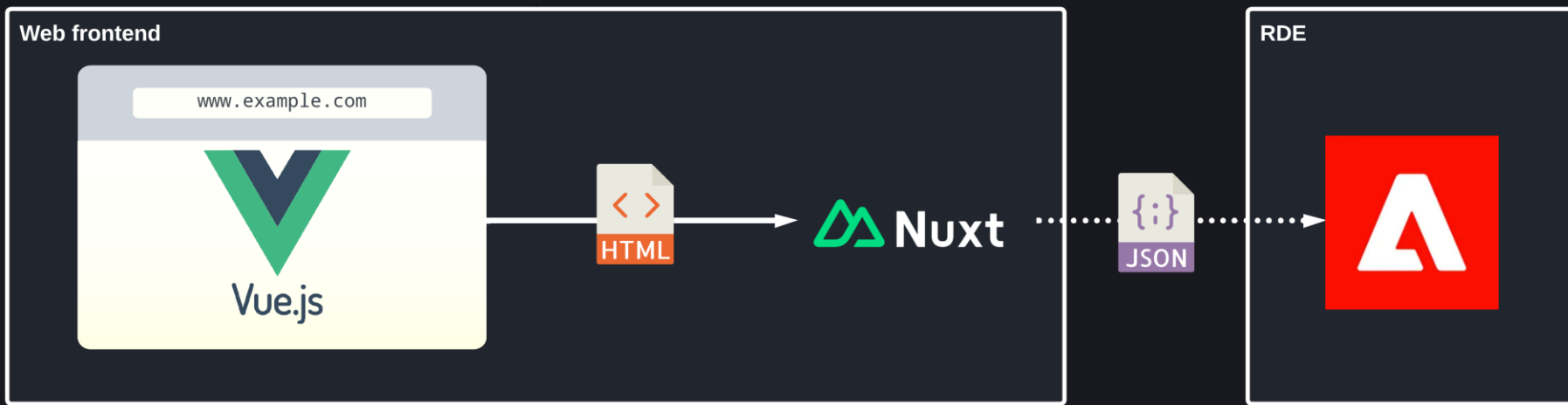
What's wrong with local AEM setup?

- Cloud vs local SDK discrepancies
- Missing surrounding services
- Behavior differences

Review apps

- New merge request == new RDE
- Streamlined testing
- Change isolation
- Hassle-free setup

Review apps



feat: RDE test

[Edit](#)[Code](#) ▾

Wadolowski, Jakub (EXTERN) requested to merge [feature/rde-test](#) into [develop](#) 38 minutes ago

[Overview](#) 1[Commits](#) 5[Pipelines](#) 6[Changes](#) 1[Add a to do](#)

0



0



Merge request pipeline [#3959794](#) waiting for manual action



Merge request pipeline waiting for manual action for [0f452a8a](#)

Test coverage 95.00% from 1 job [?](#)

Deployed to [rde/author-feature-rde-test](#) 5 minutes ago

[View app](#) 

Deployed to [rde/publish-feature-rde-test](#) 5 minutes ago

[View app](#) 

Deployed to [web-feature-rde-test](#) 4 minutes ago

[View app](#) 

0 Assignees

[Edit](#)

None - assign yourself

0 Reviewers

[Edit](#)

None - assign yourself

Labels

[Edit](#)

None

Milestone

[Edit](#)

None

Time tracking



No estimate or time spent

8 Requires 1 approval from eligible users.



Gitlab pipeline

build

✓ build-and-sonar 

test

✓ code_quality 

✓ prisma_scan 

✓ secret_detection 

✓ spotbugs-sast 

cloudmanager-sync

✓ sync-branches 7

pre-deploy

✓ rde-assignment 

deploy



✓ deploy-rde-author 

✓ deploy-rde-publish 

✓ deploy-rde-web 

⚙ remove-rde-author 

⚙ remove-rde-publish 

⚙ remove-rde-web 

reports

✓ coverage 

✓ prisma-reports 

✓ sast-report 

- Pre-provisioned RDEs
- 3 Gitlab environments per MR
 - RDE Author
 - RDE Publish
 - Web frontend
- RDE state/allocation management



Gitlab environments

Web (Vue.js)

.gitlab-ci.yml snippet

```
.gitlab-ci.yml

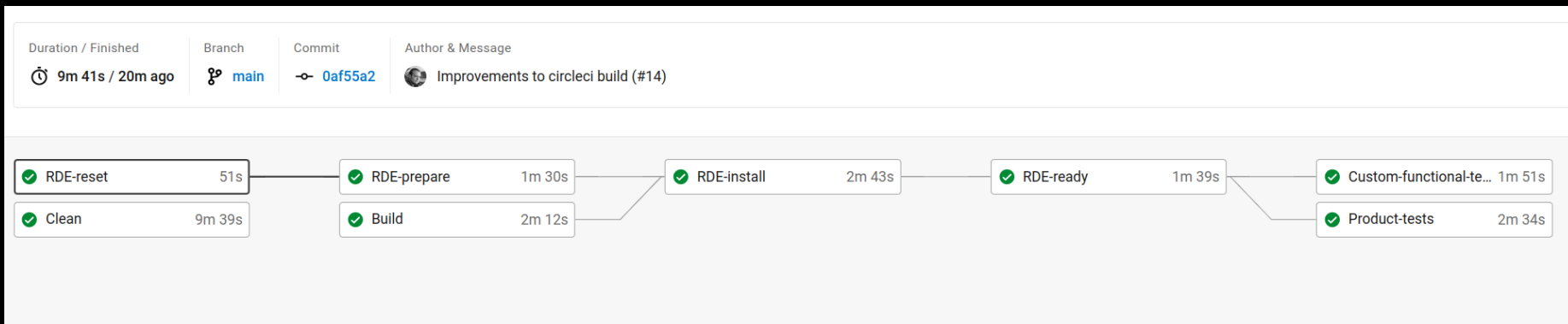
deploy-rde-author:
  stage: deploy
  needs: ['rde-assignment', 'build-and-sonar']
  environment:
    name: rde/author-${CI_COMMIT_REF_SLUG}
    deployment_tier: development
    url: https://author-p11111-e2222222.adobeaemcloud.com/libs/granite/core/content/login.html
    auto_stop_in: 4 week
    on_stop: remove-rde-author
  script:
    - "..."/>
  rules:
    - if: '$CI_PIPELINE_SOURCE == "merge_request_event"'

remove-rde-author:
  stage: deploy
  environment:
    name: rde/author-${CI_COMMIT_REF_SLUG}
    url: https://author-p11111-e2222222.adobeaemcloud.com/libs/granite/core/content/login.html
    action: stop
  script:
    - "..."/>
  rules:
    - if: '$CI_PIPELINE_SOURCE == "merge_request_event"'
      when: manual
```

Q&A

Appendix

CircleCi: overall pipeline





39s 🔍 ↗️ ⬇️

Show less

```

1 Uploading [REDACTED] 100% | ETA: 0s | 0/0 KB
2 - applying update
3 - applying update
4 - applying update
5 - retrieving update status
6 - retrieving update logs
7 #389: deploy completed for osgi-config org.apache.sling.jcr.repoinit.RepositoryInitializer-testadmin-user-add.cfg.json on author
8 ,publish - done by ***** at 2023-08-21T13:51:35.572Z
9 No logs available for this update.
10 - deleting org.apache.sling.jcr.repoinit.RepositoryInitializer-testadmin-user-add
11 - retrieving update status
12 - retrieving update logs
13 #390: delete completed for osgi-config org.apache.sling.jcr.repoinit.RepositoryInitializer-testadmin-user-add on author - done b
14 y ***** at 2023-08-21T13:51:57.624Z
15 No logs available for this update.
16 - retrieving update status
17 - retrieving update logs
18 #391: delete completed for osgi-config org.apache.sling.jcr.repoinit.RepositoryInitializer-testadmin-user-add on publish - done
19 by ***** at 2023-08-21T13:52:04.940Z
20 No logs available for this update.

```

CircleCi – Build: maven project

▼  build project

1m 45s 🔍 ↗️ ⬇️

```
#!/bin/bash -eo pipefail
# create the workspace dir
mkdir -p workspace/all
mkdir -p workspace/dispatcher

# build
mvn clean install

# copy the artifacts into a workspace location
cp -r all/target/*.zip workspace/all
cp -r dispatcher/target/*.zip workspace/dispatcher
```

Show less

```
[INFO] Installing /home/circleci/project/ui_tests/target/aem-guides-wknd-ui_tests-2.1.3-SNAPSHOT-ui-test-docker-context.tar.gz to /home/circleci/.m2/repository/com/adobe/aem-guides-wknd-ui_tests/2.1.3-SNAPSHOT/aem-guides-wknd-ui_tests-2.1.3-SNAPSHOT-ui-test-docker-context.tar.gz
[INFO] -----
[INFO] Reactor Summary for WKND Sites Project - Reactor Project 2.1.3-SNAPSHOT:
[INFO]
[INFO] WKND Sites Project - Reactor Project ..... SUCCESS [ 0.455 s]
[INFO] WKND Sites Project - Core ..... SUCCESS [ 8.571 s]
[INFO] WKND Sites Project - UI Frontend ..... SUCCESS [ 27.917 s]
[INFO] WKND Sites Project - UI apps structure ..... SUCCESS [ 0.876 s]
[INFO] WKND Sites Project - UI apps ..... SUCCESS [ 4.370 s]
[INFO] WKND Sites Project - UI content ..... SUCCESS [ 1.993 s]
[INFO] WKND Sites Project - UI config ..... SUCCESS [ 0.083 s]
[INFO] WKND Sites Project - UI sample content ..... SUCCESS [ 0.708 s]
[INFO] WKND Sites Project - All ..... SUCCESS [ 48.705 s]
[INFO] WKND Sites Project - Integration Tests ..... SUCCESS [ 8.150 s]
[INFO] WKND Sites Project - Dispatcher ..... SUCCESS [ 0.209 s]
[INFO] WKND Sites Project - UI Tests ..... SUCCESS [ 0.189 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:43 min
[INFO] Finished at: 2023-08-21T13:51:52Z
[INFO] -----
```

▶  Persisting to workspace

6s 🔍 ↗️ ⬇️



CircleCi – RDE Install

38

CircleCi – RDE ready

▼  Wait until RDE endpoint is ready

1m 31s



```
#!/bin/bash -eo pipefail
sleep 10
./wait-endpoint.sh "${PUBLISH_URL}"
```

```
1  Wait for URLs: *****
2
3  Testing: *****
4  Waiting for *****
5  Waiting for *****
6  Waiting for *****
7  Waiting for *****
8  Waiting for *****
9  0
10 OK: *****
11
12 SUCCESSFUL
13
```

