

Gem Session 29 August – Pattern detector, Testing

Arek Kita, Andrei Dulvac

Bē
Jon Noorlander

Agenda

Upgrade Gem sessions

Pattern detector

Enforcing backwards compatibility

Summary

Q&A



Upgrade Gem Sessions

Bē

Jon Noorlander

Upgrade Gem Sessions

- 22 August – Backwards compatibility, Surface API - DONE
- 29 August – Pattern detection, Testing– this one
- 20 September – Mockingjay 6.4
- Sometime in October – Upgrade/Surface API/Backwards compatibility



Pattern detector

Arek Kita

Bē

Jon Noorlander

Pattern detector

DEMO



Backwards compatibility enforcement

Andrei Dulvac

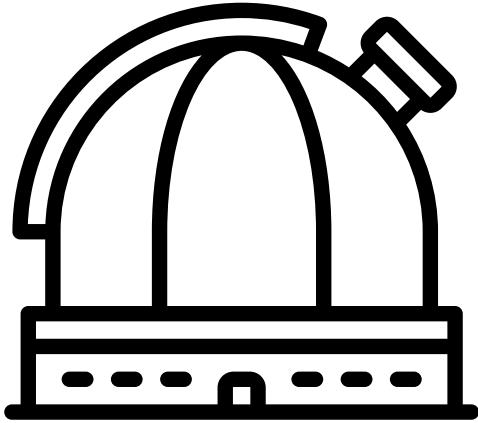
Bē

Jon Noorlander

Testing

- Enforcing backwards compatibility
- Testing compat packages

Enforcing backwards compatibility



- ContentModelIT
 - Prevent unintentional breaking changes
 - Make exception handling and moves to compat packages traceable
- BundleVersion Reference Check
 - Prevent accidental introduction of major version bumps (e.g. 3rd party libs)
 - Make exception handling and moves to compat packages traceable
- Those policies and checks are generic and exceptions are expected:
 - **False negative: RTC to clarify why breach creates no compatibility issue**
 - **Unavoidable breaking change: JM approval**

Testing compat packages

- Multiple compat packages add big testing complexity => ONE compat package

GOAL	Testing Setup	When
Test if 6.4 is compatible with 6.3	6.4 + compat package + 6.3 evergreen tests	Phase 1 – targeted L12 (depending on HW readiness)
Test if 6.4 features work in compat mode	6.4 + compat package + subset of 6.4 evergreen tests	Phase 2 – TBD

Testing

DEMO

Current run results (#30)

- 6.4 + compat package
 - <https://jenkins-evergreen.bsl.eur.adobe.com/job/bc-poc-it-tar-master-pr/lastCompletedBuild/testReport/>
 - 1490 tests, 477 failed (exception list + newly introduced issues)

Configuration Name	Duration	All	Failed	Skipped
featureGroup=bc-compile,jdk=JDK8,label=Evergreen	21 min	14	0	0
featureGroup=bc-projects,jdk=JDK8,label=Evergreen	5 min 1 sec	40	0	21
featureGroup=bc-mcm,jdk=JDK8,label=Evergreen	1 min 2 sec	27	0	12
featureGroup=bc-foundation,jdk=JDK8,label=Evergreen	6 min 17 sec	232	0	10
featureGroup=bc-security,jdk=JDK8,label=Evergreen	7 min 38 sec	113	1	7
featureGroup=bc-msm,jdk=JDK8,label=Evergreen	4 min 49 sec	165	31	17
featureGroup=bc-commerce,jdk=JDK8,label=Evergreen	1 min 51 sec	330	253	69
featureGroup=bc-platform,jdk=JDK8,label=Evergreen	6 min 38 sec	119	0	9
featureGroup=bc-wcm,jdk=JDK8,label=Evergreen	4 min 56 sec	163	3	11
featureGroup=bc-sites,jdk=JDK8,label=Evergreen	2 min 44 sec	132	2	26
featureGroup=bc-security-privileges,jdk=JDK8,label=Evergreen	7 min 19 sec	58	0	5
featureGroup=bc-tagging,jdk=JDK8,label=Evergreen	5 min 4 sec	97	0	0

What to do when the job fails

- Always check the BC evergreen job when doing a PR – for now it is non blocking
- If there is a test failure, investigate the reason, run test locally
- If not fixable, open BLOCKING ISSUE
- Add exception to the list of BC exceptions

Next steps

- Deploying on new hardware
- Validating exception list with PM
- Create load blocking Jira issues for all the failures
- Enabling blocking builds
 - **Targeted for M4 (L12) – Testing 6.4 + compat package backwards compatibility with 6.3**
- Adding support for additional testing configurations
- Teams to add BC tests in evergreen sprouts

Q&A

Q&A



Summary

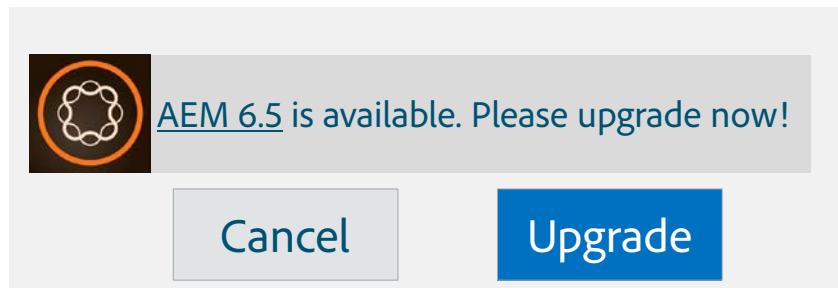
Costin Genescu

Bē

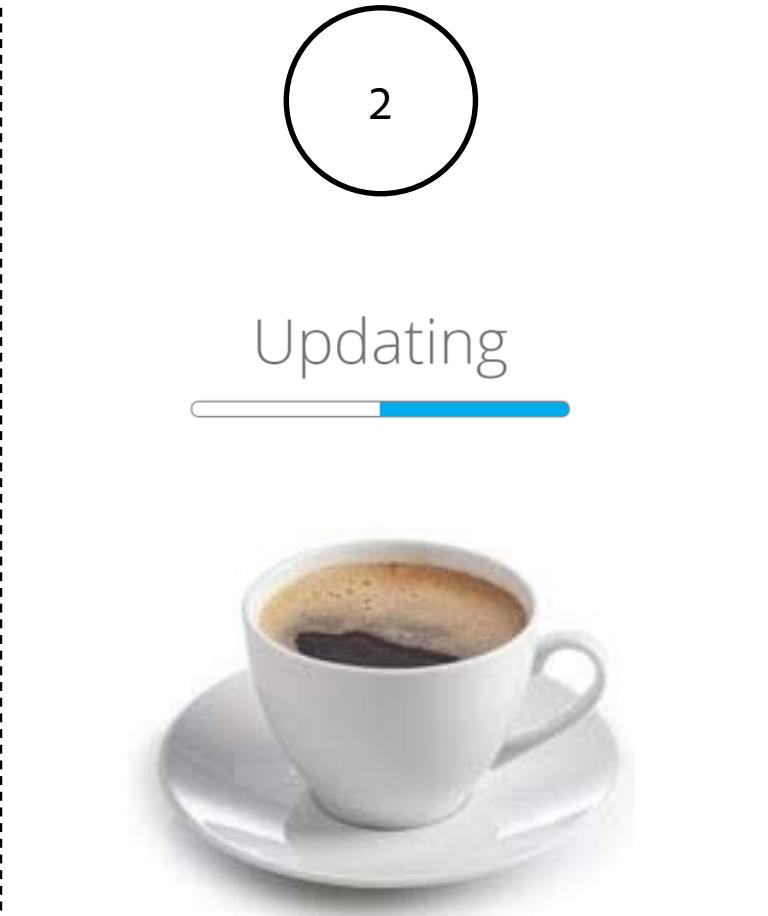
Jon Noorlander

Upgrade vision

1



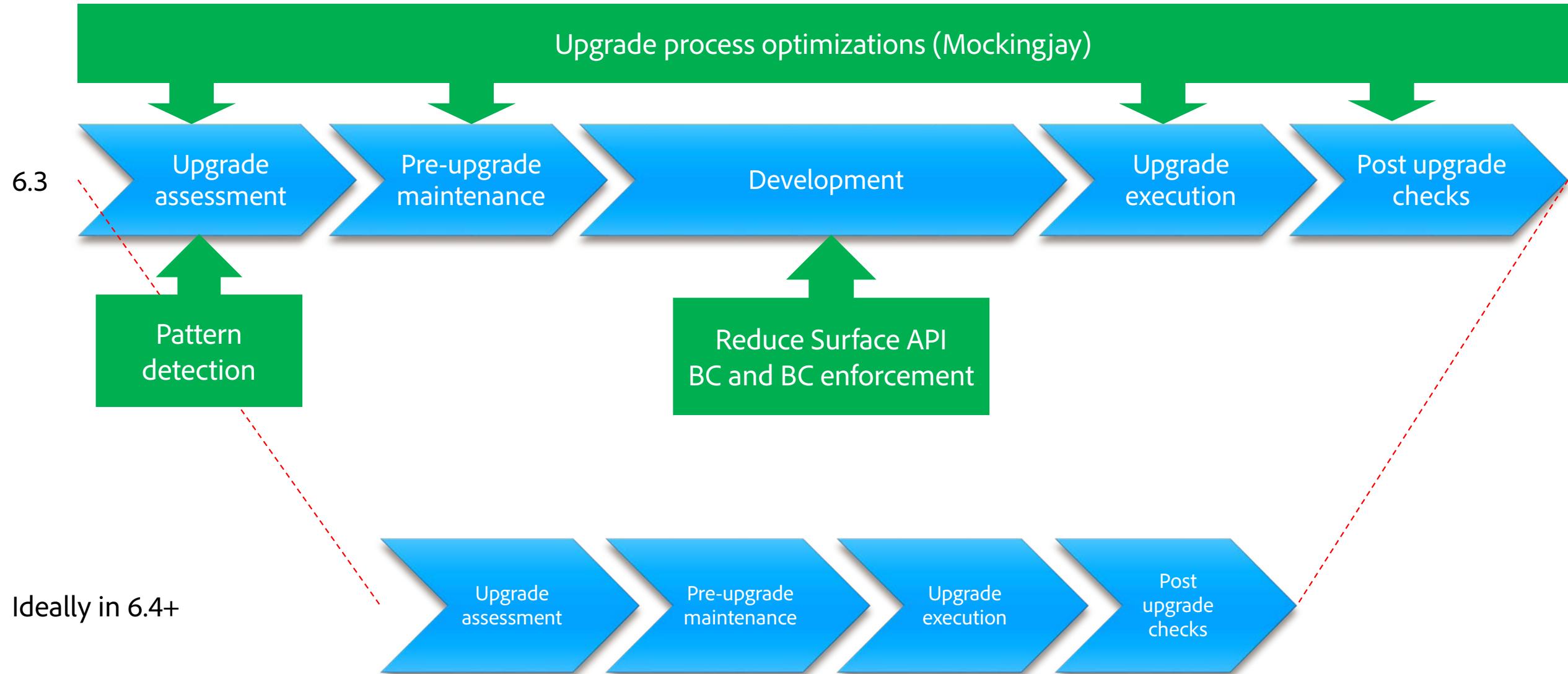
2



3



Upgrade project scope



Deliverables

Jira	Deliverables	Teams work required
CQ-4194070 Framework for zero touch upgrade	Documentation for AEM surface API Compat Package and lifecycle KPIs for upgrade process	Surface API Backwards compatibility
CQ-4194071 Enforce backwards compatibility for AEM feature developers	Enable automatic validation of backward compatibility Enable Engineers to detect breaking changes at build time Enable a framework for the E2E validation of the customer upgrade	Add tests for enforcing the surface API QA team must create the framework for testing compat packages
CQ-85533 Pattern-Detector	Tool that detects patterns that are known to break upgrades The pattern library	Contribute to the pattern library

Mockingjay 6.4

Jira	Deliverables	Teams work required
CQ-4196505 Regular upgrade testing	20 Customer upgraded in the Mockinjay program	60 engineers will be asked to contribute

More on 20 September GEM Session

More details

- Costin Genescu genescu@adobe.com
- Dominik Suess suess@adobe.com
- Arek Kita arkadius@adobe.com
- Vlad Petcu petcu@adobe.com
- Andrei Dulvac adulvac@adobe.com
- <https://wiki.corp.adobe.com/display/WEM/AEM+Upgrades>



Adobe

MAKE IT AN EXPERIENCE