



**DEVELOPERS LIVE**



# **AEM as a Cloud Service Local SDK Build Analysers**

**Brian Chaikelson | Principal Product Manager**

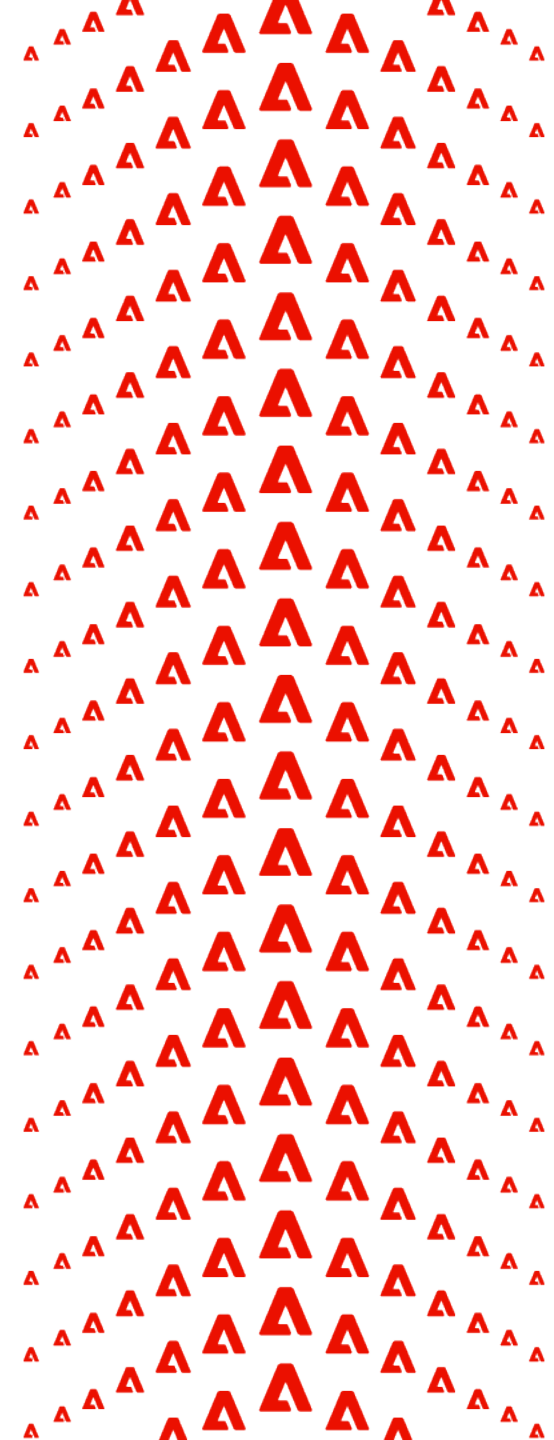
**David Bosschaert | Senior Software Engineer**

**Karl Pauls | Computer Scientist**

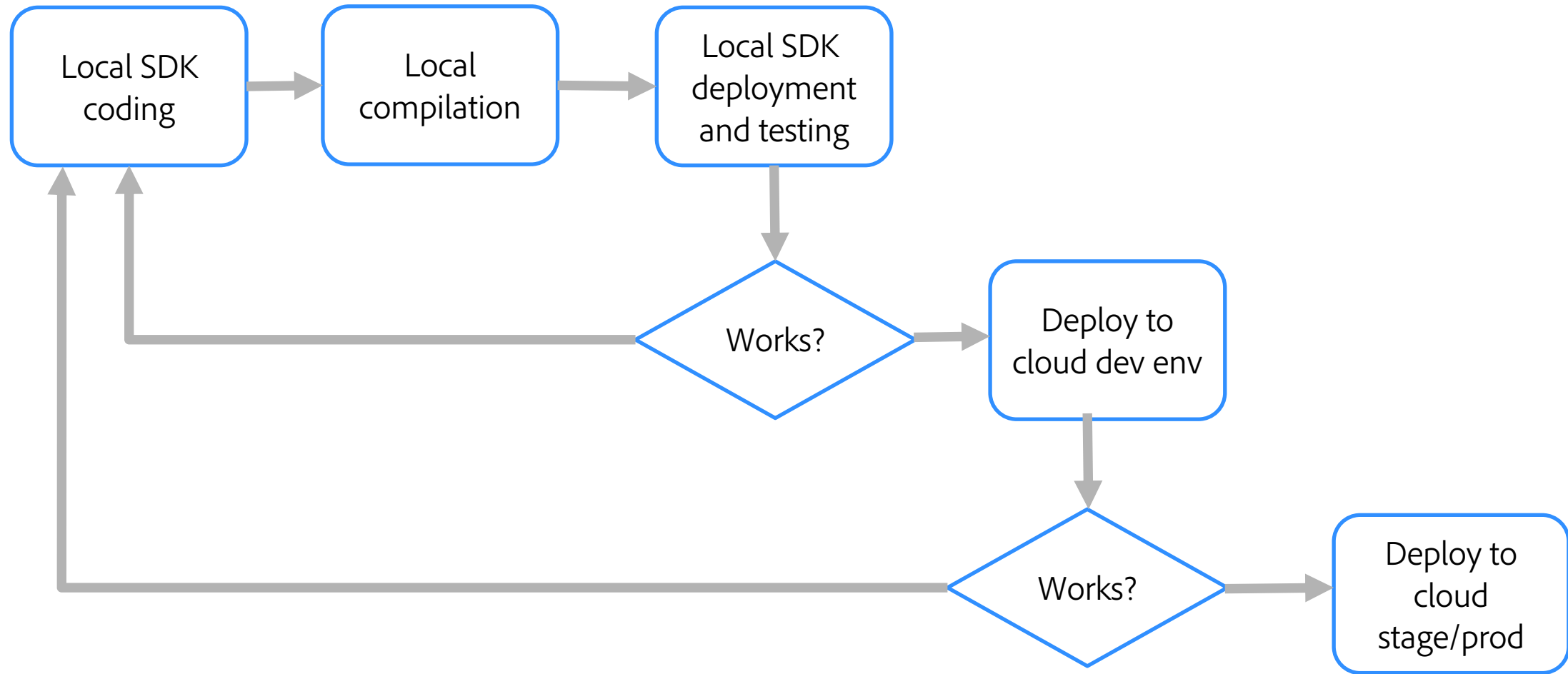


# Agenda

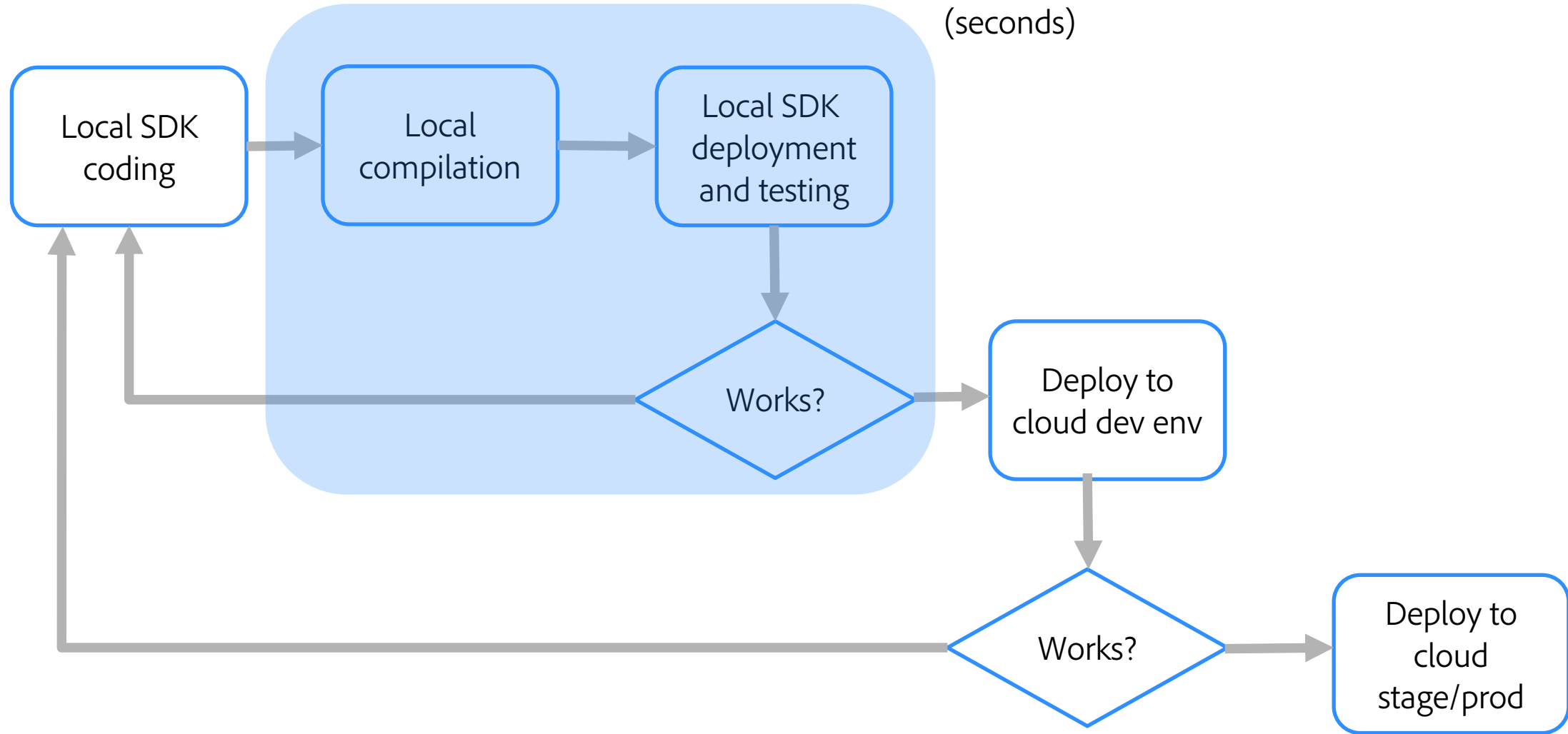
|                  |                           |
|------------------|---------------------------|
| Brian Chaikelson | Background and motivation |
| David Bosschaert | Analyser details          |
| Karl Pauls       | Demo                      |
|                  | Q&A                       |



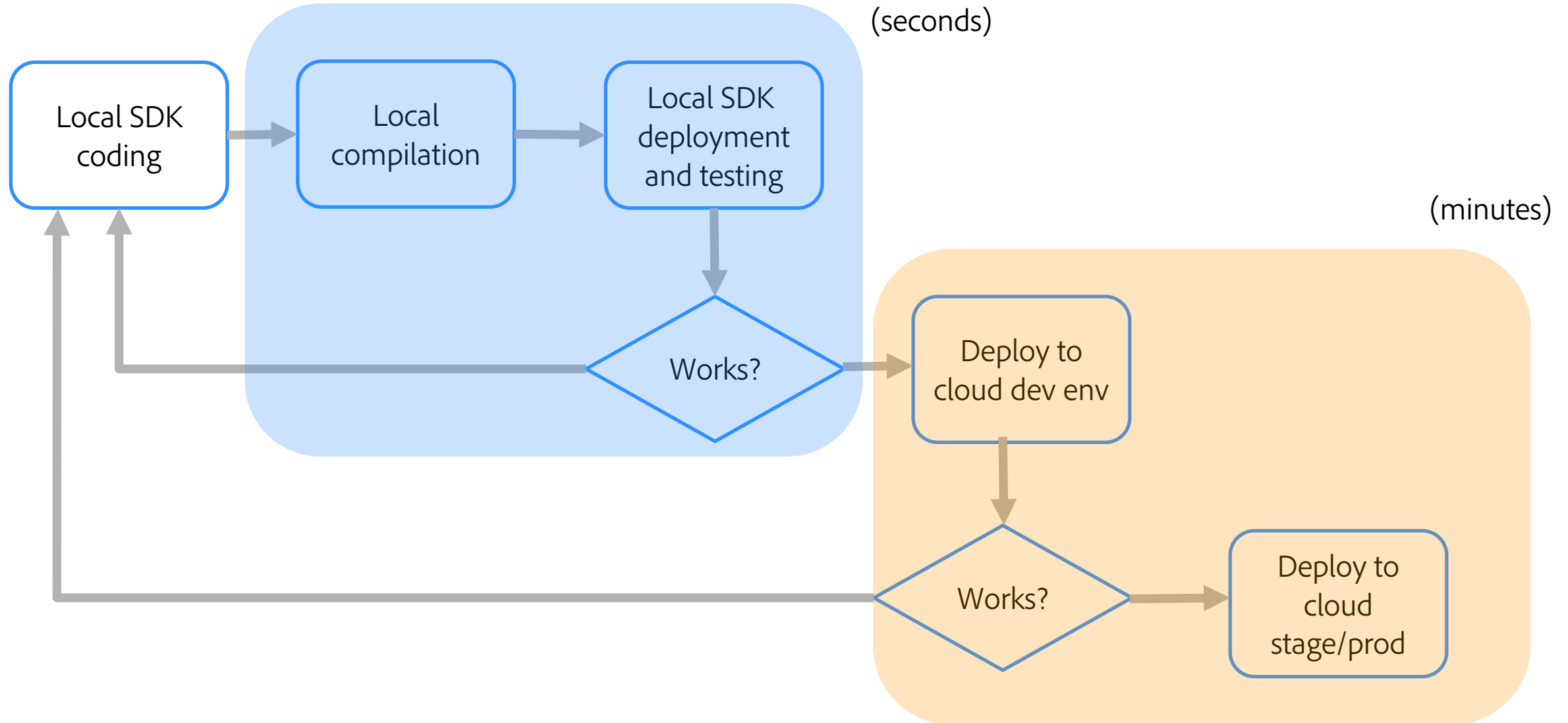
# AEM as a Cloud Service Developer Flow



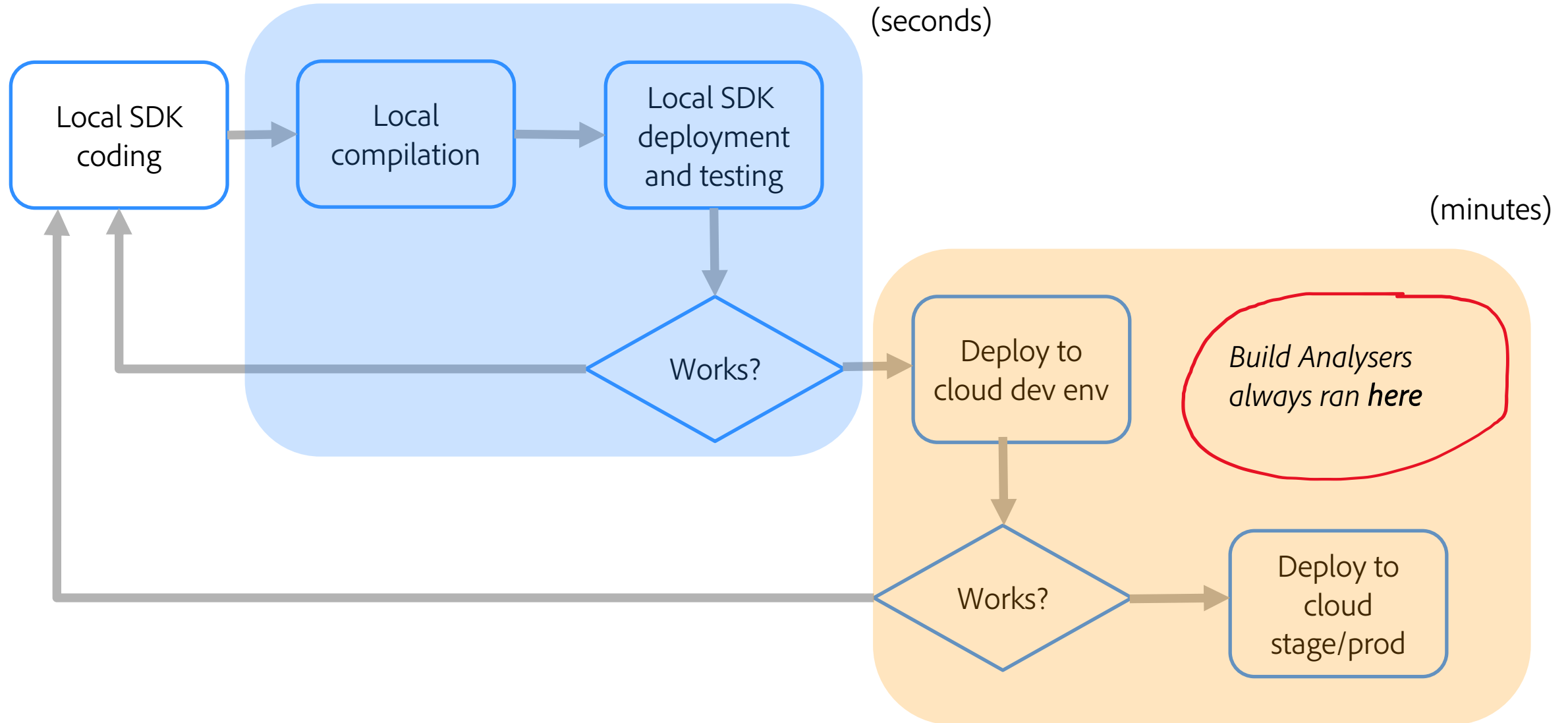
# AEM as a Cloud Service Developer Flow



# AEM as a Cloud Service Developer Flow



# AEM as a Cloud Service Developer Flow



# AEM as a Cloud Service Developer Flow

CI/CD orchestrated via Cloud Manager,  
with steps and quality gates:

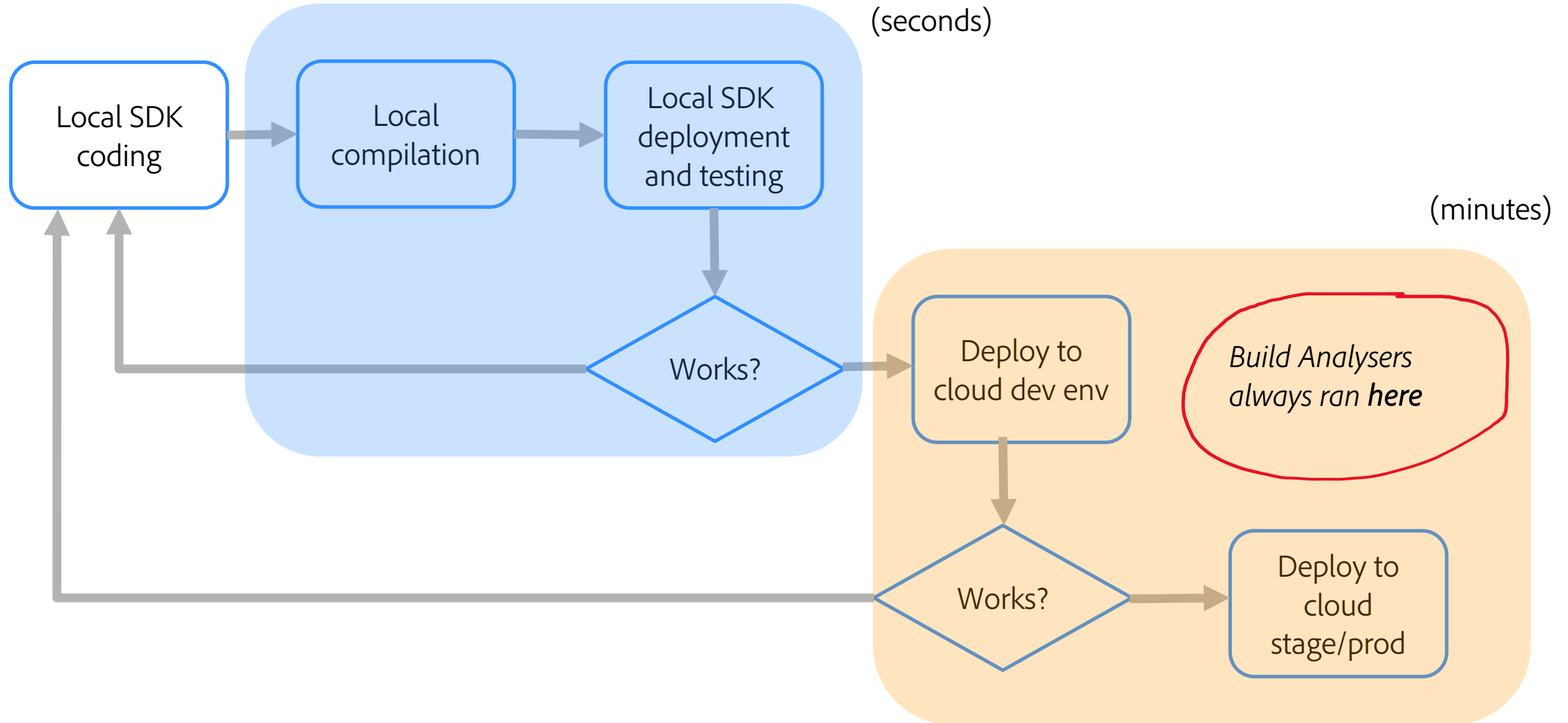
1. Validation
2. Build & Unit Testing
3. Code Scanning
4. Build Images
5. Deploy

Analyser looks for issues in the maven  
project.

The screenshot displays the Adobe Cloud Manager interface for a "Non-Production Pipeline". The pipeline is marked as "Finished". The left sidebar shows the pipeline structure with "Build and Code Scanning" selected. The main content area details the following steps:

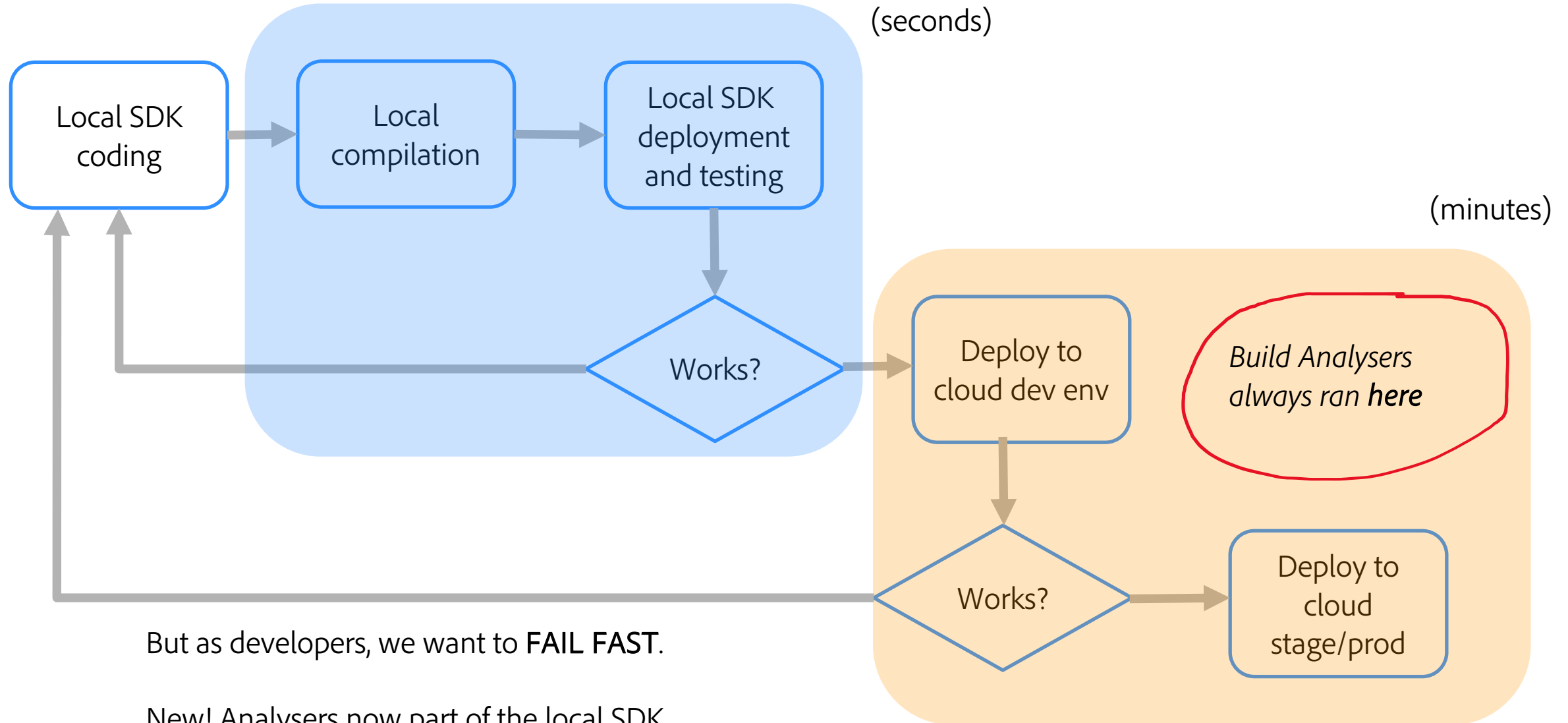
- Validation**: Your Pipeline has been verified against a set of pre-defined rules. Status: **Passed** (Finished: January 15, 2021 3:50:56 PM PST).
- Build & Unit Testing**: Your application code has been tested for quality and compiled into build artifacts. BRANCH: wknd. Status: **Succeeded** (Finished: January 15, 2021 4:01:15 PM PST). Includes a "Download Log" button.
- Code Scanning**: Your application code has been verified against a selected set of rules used for AEM applications. Status: **Passed** (Finished: January 15, 2021 4:01:22 PM PST). Includes buttons for "Review Summary", "Download Details", and "Download Log".
- Build Images**: Your built artifacts will be combined with the base AEM 2020.12.4687.20201218T031154Z to produce an image runnable in the cloud. AEM RELEASE: 2020.12.4687.20201218T031154Z. Status: **Completed** (Finished: January 15, 2021 4:20:22 PM PST). Includes a "Download Log" button.
- NON-PROD DEPLOYMENT**: Includes a "Deploy to Dev" step.

# AEM as a Cloud Service Developer Flow





# AEM as a Cloud Service Developer Flow



But as developers, we want to **FAIL FAST**.

New! Analysers now part of the local SDK.

# AEM Analyser Maven Plugin

Runs the AEMaaCS analysers in your local builds

| Analyser                      | Function   |
|-------------------------------|--|
| api-regions-exportsimports    | Checks if all OSGi bundles have their Import-Package declarations satisfied by the Export-Package declaration of bundles   |
| bundle-nativecode             | Ensures that bundles don't contain any native code   |
| api-regions-crossfeature-dups | Checks that customer OSGi bundles don't have Export-Package declarations that override AEM as a Cloud Service's public API |
| repointit                     | Checks the syntax of all repointit sections  |
| ... other ...                 | Additional analysers perform internal validations  |

# Analyser Framework

- Local Analysers
  - executed via the aemanalyser-maven-plugin [0]
  - Simple: create a new submodule to execute the analysers
- Analyser framework is open source [1]
- You create your own analysers too and add them 😊

[0] <https://github.com/adobe/aemanalyser-maven-plugin>

[1] <https://github.com/apache/sling-org-apache-sling-feature-analyser>



Image: Adobe Stock

# Enabling the local analysers

```
<project>
  ...
  <packaging>aem-analyse</packaging>

  <build>
    <plugins>
      <plugin>
        <groupId>com.adobe.aem</groupId>
        <artifactId>aemanalyser-maven-plugin</artifactId>
        <extensions>true</extensions>
      </plugin>
    </plugins>
  </build>

  <dependencies>
    <dependency>
      <groupId>com.mysite</groupId>
      <artifactId>myproject.all</artifactId>
      <type>zip</type>
    </dependency>
  </dependencies>
</project>
```

Make sure com.adobe.aem:aem-sdk-api is available in dependency management - AEM Archetype does this also for you.

**Demo**



# Takeaways and Q&A

Use local SDK build analysers to detect problems in your maven project locally to FAIL FAST in seconds, rather than deploying to Cloud, which takes minutes.

Include them in your pom.xml today!

Public documentation: <http://adobe.ly/39RpVFs>

