APACHE SLING & FRIENDS TECH MEETUP
BERLIN, 25-27 SEPTEMBER 2017

Context-Aware Configuration in AEM

Stefan Seifert, pro!vision GmbH

About the Speaker

- AEM Developer
- Apache Sling PMC
- CTO of pro!vision GmbH

Stefan Seifert

PRO!VISION
SOFTWARE CRAFTSMANSHIP

https://www.pro-vision.de

Recap:
Sling Context-Aware Configuration

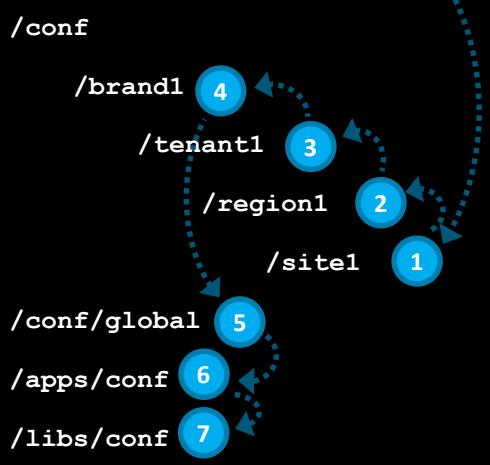
Context-Aware Configuration (CAConfig)

- Context-aware configurations
  are configurations that are
  related to a content resource or a resource tree,
  e.g. a web site or a tenant site.

- != System Configuration

Configuration lookup

```
/content
    /tenant1
      @sling:configRef = "/conf/brand1/tenant1"
        /region1
          @sling:configRef = "/conf/brand1/tenant1/region1"
            /site1
              @sling:configRef = "/conf/brand1/tenant1/region1/site1"
```



Resource lookup order for configuration hierarchies

Fallback to `/conf/global` and `/apps/conf` and `/libs/conf`

Core Features

- Singleton Configurations
- Configuration collections
- Describe configuration via Java class
- Collection and property inheritance
- SPI to adapt to you needs

Sling Context-Aware Configuration

For more details see:

Talk at adaptTo() 2016

https://adapt.to/2016/en/schedule/sling-context-aware-configuration.html

Deploy CAConfig in AEM

Deploy in AEM

- CAConfig is already in included in AEM 6.3
  - But you should update to latest bundles
- You can also deploy CAConfig to AEM 6.1 or AEM 6.2
  - Additional OSGi configuration needed

Latest Sling CAConfig Bundles

Bundle	Version
Apache Sling Context-Aware Configuration API `org.apache.sling.caconfig.api`	1.1.0
Apache Sling Context-Aware Configuration SPI `org.apache.sling.caconfig.spi`	1.3.2
Apache Sling Context-Aware Configuration Implementation `org.apache.sling.caconfig.impl`	1.4.4
Apache Johnzon Wrapper Library `org.apache.sling.commons.johnzon`	1.1.0

It is safe to update these bundles that come with AEM to the latest versions.

AEM-specific OSGi Configuration

Support for `cq:Page`/`cq:PageContent` nodes

```
org.apache.sling.caconfig.resource.impl.def.DefaultContextPathStrategy
    configRefResourceNames=["jcr:content"]
```

Ignore properties with `jcr:` and `cq:` prefix

```
org.apache.sling.caconfig.management.impl.ConfigurationManagementSettingsImpl
    ignorePropertyNameRegex=["^(jcr|cq):.+$"]
```

Full instructions:

http://wcm.io/caconfig/deploy-configure-caconfig-in-aem.html

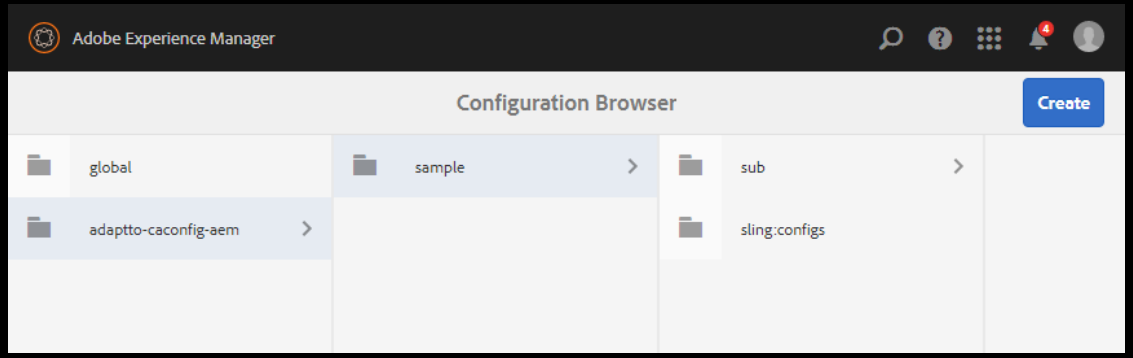
adaptTo()

AEM CAConfig OOTB

OOTB CAConfig Support in AEM 6.3

- ✓ Reading from `cq:Page` nodes at `/conf`
- ✗ Writing to `cq:Page` nodes at `/conf`
- ✗ Declarative definition of contexts
- ✗ Configuration Editor GUI
- ✗ Activating Configuration to Publish via GUI

AEM Configuration Browser



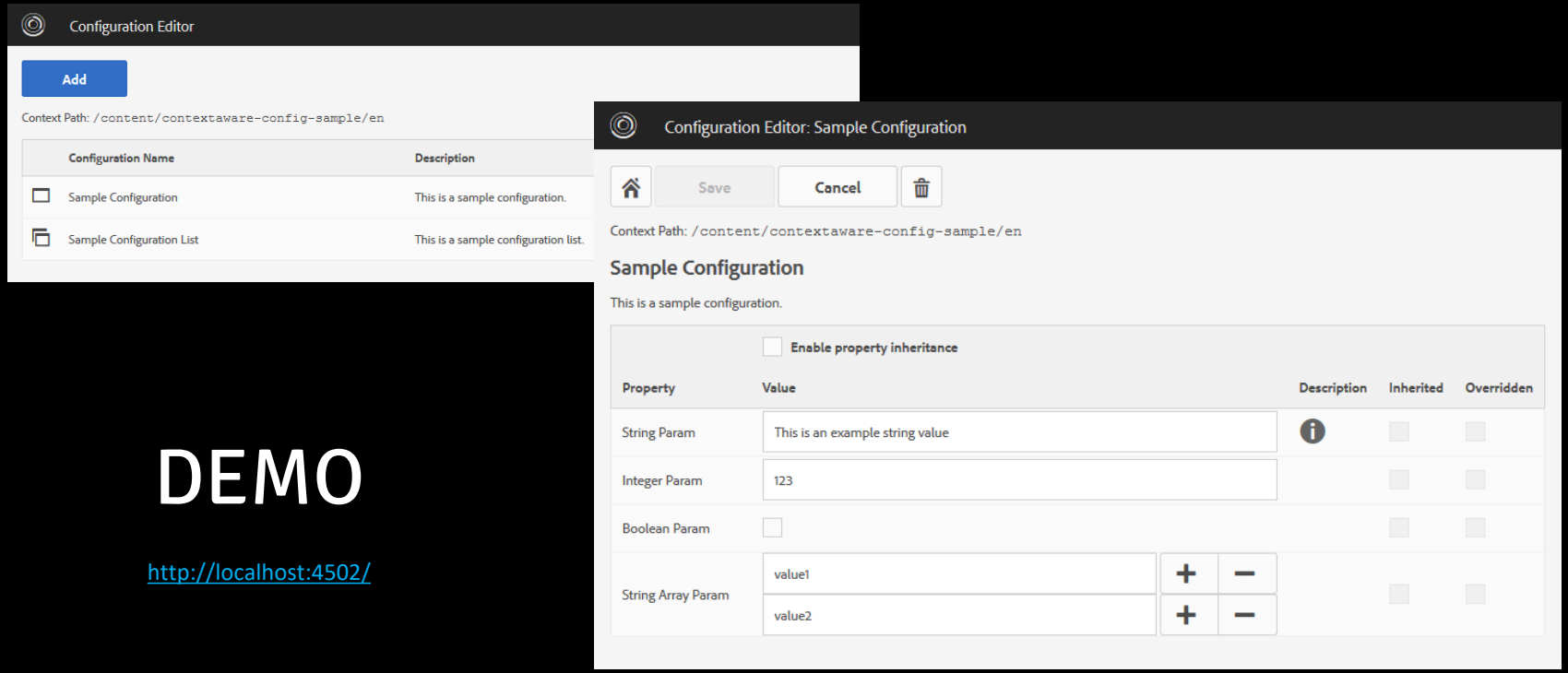
- AEM contains a “configuration browser”
- But this only allows to browse `/conf` folders, not the configurations in them

AEM ConfMgr

- Introduced in AEM 6.1
- Since AEM 6.3 only a thin wrapper around the CAConfig API
- Applications should migrate from ConfMgr API to CAConfig API

Managing Configurations in AEM

Introducing wcm.io CAConfig Editor



DEMO

http://localhost:4502/

wcm.io CAConfig Editor Features

- Edit singleton and configuration collections
- Edit widget depending on data type
- Control collection and property inheritance
- Custom widgets like path browser

http://wcm.io/caconfig/editor/

Placing config editor page

- The configuration editor is created as AEM page within the context
- Reads and writes the config from `/conf`

```
/content
    /mysite
      @sling:configRef = "/conf/mysite"
        /tools
           /config
/conf
    /mysite
        /sling:configs
            /x.y.z.MyConfig
              @param1 = "value1"
```

Configuration editor page

Configuration stored at `/conf`

Context Path Strategies for AEM

Default Context Path Strategy

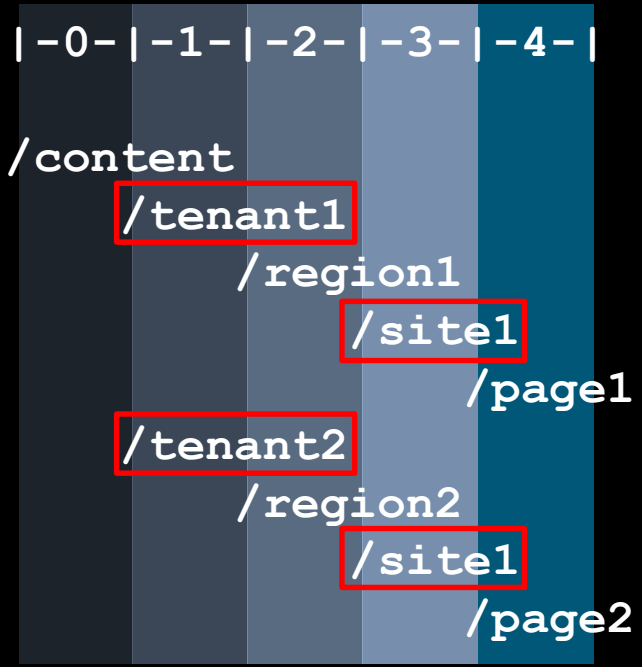
- CAConfig context is defined by setting a `sling:configRef` property on each root page – pointing to a path at `/conf`
- This gets tedious if you have a huge number of sites/contexts following a common path convention

Introducing wcm.io CAConfig Extensions

- wcm.io CAConfig Extensions provide Context Path Strategies with a declarative approach
- Use OSGi configuration to declare the strategy
- Contexts are defined automatically

http://wcm.io/caconfig/extensions/

Absolut Parent Context Path Strategy



- A fixed set of “absolute parent” path levels is used to define the context roots
- Example: Levels **1, 3**

Root Template Context Path Strategy

```
|-0-|-1-|-2-|-3-|-4-|
/content
    /tenant1 <Structure Template>
        /region1 <Structure Template>
            /site1 <Homepage Template>
                /page1 <Content Template>
```

- Pages using a configurable template are detected as context root. Parents may define nested contexts.

Derive config paths

- Define a rule how to match config path to content path

contextPathRegex = `"^/content(/.+)$"`

configPathPatterns = `["/conf$1"]`

Context root path = `/content/tenant1/region1/site1`

Derived configuration path = `/conf/tenant1/region1/site1`

Persistence Strategies for AEM

Default persistence strategy

- By default, configurations are stored as `nt:unstructured` nodes below `/conf`
- wcm.io CAConfig Extension provides alternative strategies using `cq:Page` nodes

AEM Page Persistence Strategy

- Stores configurations in `cq:Page/jcr:content`
- Makes it easier to replicate them to publish individually
- Uses similar content model as AEM ConfMgr

Tools Config Page Persistence Strategy

- Stores configurations in `tools/config` pages as **part of the content**, and **not** below `/conf`

Advantages:

- Configuration can be packaged or replicated easily together with content
- Configuration can be activated, versioned etc. directly from Author GUI

Disadvantages:

- Configuration cannot be easily protected via ACLs
- Mixes content and configuration

Using Override Providers

Use Case

- You use CAConfig to define the “externalization” base URLs for each site
  - e.g. `siteUrl = "http://mydomain.com"`
- You have some “feature flag” parameters to enable/disable features in country sites
  - e.g. `carConfigurator = true|false`

Scenario

- You want to deploy the whole production content to your test system
  - Set all Site URLs to a test domain
- You want to disable a feature flag on all your live websites due to a temporary problem
  - Switch it off globally, preserve previous state

Use CAConfig Override

- Deploy overrides e.g. via OSGi configuration

Overrides whole config with new `siteUrl` on all paths

```
x.y.z.MyConfig={'siteUrl':'http://mytest.com'}
```

Overrides single feature flag for all sites of `tenant1`

```
[/content/tenant1]x.y.z.MyConfig/carConfigurator=false
```

Unit Test Mock Support

Use AEM Mocks

- Use wcm.io AEM Mocks with plugins

```
<dependency>
  <groupId>io.wcm</groupId>
  <artifactId>io.wcm.testing.aem-mock</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>io.wcm</groupId>
  <artifactId>io.wcm.testing.wcm-io-mock.caconfig</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.apache.sling</groupId>
  <artifactId>org.apache.sling.testing.caconfig-mock-plugin</artifactId>
  <scope>test</scope>
</dependency>
```

Unit Test Example

```
import static io.wcm.testing.mock.wcmio.caconfig.ContextPlugins.WCMIO_CACONFIG;
import static org.apache.sling.testing.mock.caconfig.ContextPlugins.CACONFIG;

public class MyTest {

  @Rule
  public AemContext context = new AemContextBuilder()
      .plugin(CACONFIG)
      .plugin(WCMIO_CACONFIG)
      .build();

  @Before
  public void setUp() {
    // register configuration annotation class
    MockContextAwareConfig.registerAnnotationPackages(context, "com.myapp.config");

    // shortcut for registering a context path strategy for unit test
    MockCAConfig.contextPathStrategyRootTemplate(context, "/apps/myapp/templates/home");
  }

  ...
}
```

Conclusion

CAConfig with the help of wcm.io

- ✓ Reading from `cq:Page` nodes at `/conf`
- ✓ Writing to `cq:Page` nodes at `/conf`
- ✓ Declarative definition of contexts
- ✓ Configuration Editor GUI
- ✗ Activating Configuration to Publish via GUI

Support planned

How to deploy

- Include these bundles in your deployment

  `io.wcm:io.wcm.caconfig.extensions`

  `io.wcm:io.wcm.caconfig.editor`

- Convenience package for CAConfig Editor

  `io.wcm:io.wcm.caconfig.editor.package`

References – wcm.io

- wcm.io Context-Aware Configuration
  http://wcm.io/caconfig/
- Training Material
  http://training.wcm.io/caconfig/
- wcm.io Community
  http://wcm.io/contribute.html

Further References

- Demo project
  https://github.com/adaptto/2017-context-aware-configuration-aem
- Sling Documentation
  https://sling.apache.org/documentation/bundles/context-aware-configuration/context-aware-configuration.html
- Talk from adaptTo() 2016
  https://adapt.to/2016/en/schedule/sling-context-aware-configuration.html