

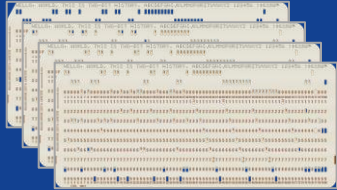
DITA and Markdown and Docker... Oh my

John Kirkilis

User Assistance Tools - Solutions Architect

Nokia Cloud and Network Services

Background



- Motorola → Freescale → NXP Semiconductor
 - Unstructured FrameMaker Writer for multi-core network SoCs
 - Migration of all content to DITA
 - XSL Development for DSP documentation
 - Oxygen customization and deployment
-
- Nokia
 - Since July 2018
 - Solutions Architect
 - Harmonizing on a single DITA and DITA-OT framework (within CNS)



Goals

- Share how we've attempted to scale up DITA and DITA-OT's customization and production in one of Nokia's three main business groups.
- Learn how others have addressed similar issues
- Meet Y'all
- Share challenges and opportunities
- Brainstorm how to strengthen the OT and LwDITA projects
 - Materially
 - Financially
- Acknowledge David Bertalan for CI maturation and TDD evangelism

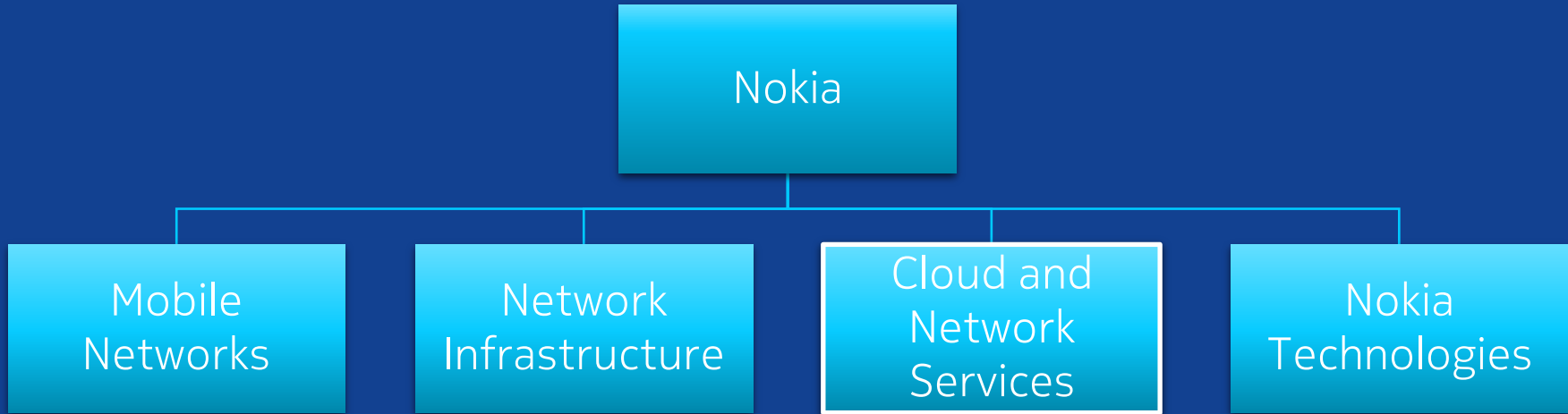
Linear Space-Time

- “Why didn’t you just use the yada, yada, yada?”
- Some things were implemented by us because:
 - specific needs not available elsewhere at the time
 - we didn't know it existed
 - we had dependencies with legacy software
 - urgent need with narrowly defined constraints
 - heavy emphasis on FOSS
 - developer skill set variations
- Caveat – may need to be intentionally vague
 - don't know
 - don't recall
 - can't say

Agenda

- Central Continuous Integration Cloud
- Tower of Babel -> Harmonized Platform based on latest DITA and DITA-OT
- Writer's Workstation Tooling – Oxygen compatibility with CI builds
- Deliverable and Publication configuration – Pre-OT Project files
- Git, Jenkins, Artifactory
- Jenkins Shared Library
- Schematron integration in CI pipeline

One business group's experiences



Central CI Farm

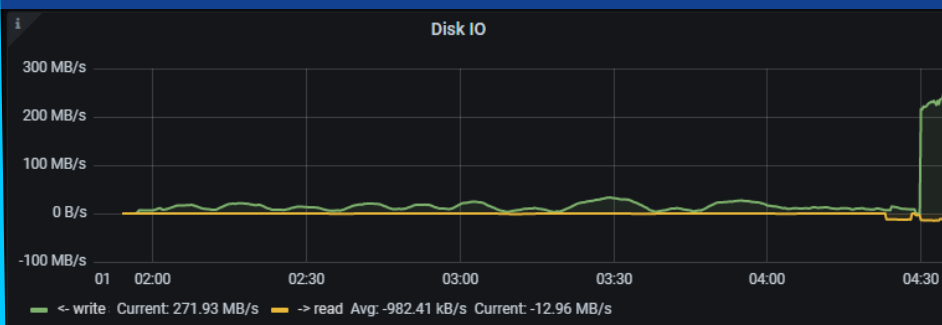
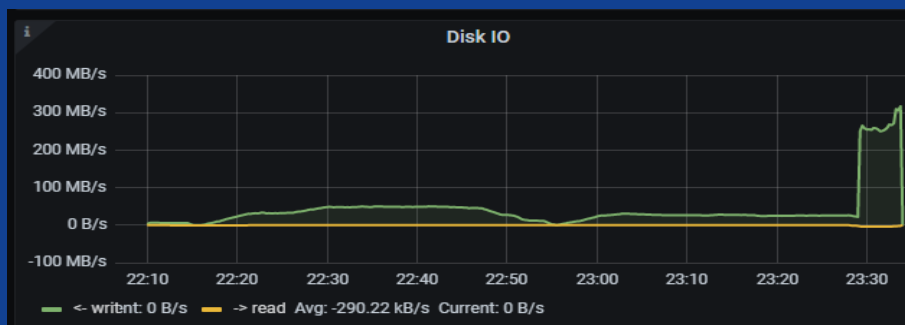
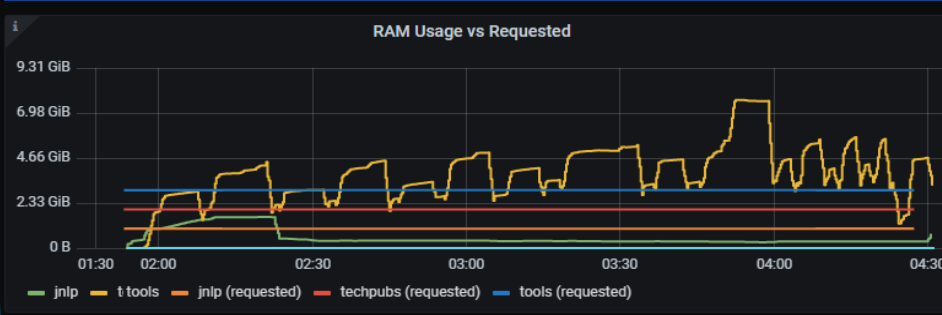
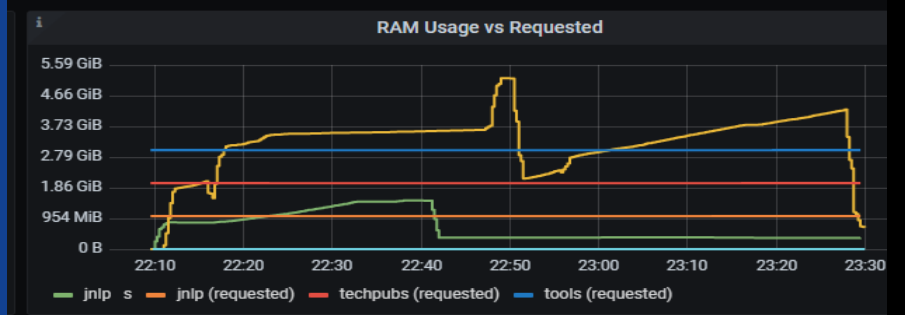
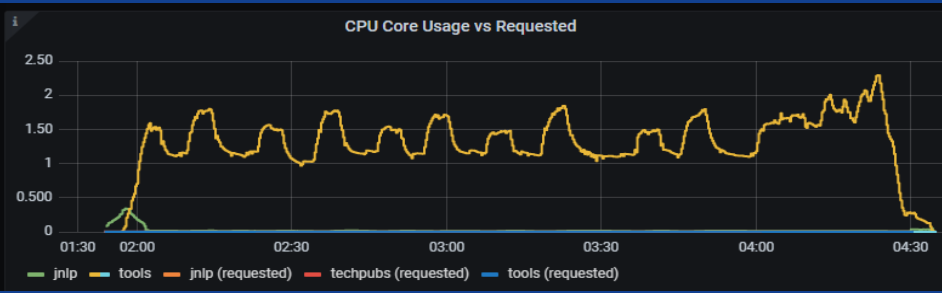
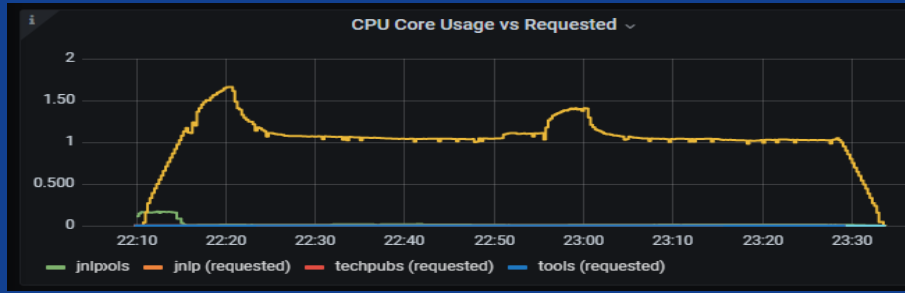
- Auto-Scaling cloud compute platform based on Kubernetes & Jenkins
- Supports R&D product software builds, UA Dev software builds (OT), Writer Content Builds (OT)
- Max peak approx. **1000** concurrent builds
- Typical daily peak approx. **500** concurrent builds(executors), baseline is never less than **200** concurrent builds.

- Limited observation of Jenkins monitoring of currently running document builds for a week never went beyond **8** concurrent TechPubs builds
- Working on logging usage via Artifactory-Jenkins plugin API in near future

Central CI Farm

2 week sample

Statistic	Value
Product docsets built	37 (out of 64)
Longest build duration average for a docset	498.81 minutes
Shortest build duration average for a docset	3.97 minutes
Average duration of all builds	30.29 minutes
Median duration of all builds	14.29 minutes
Number of writers Most of which are in U.S., Canada, Portugal, Finland, Greece, Hungary, India	approx. 120-150



Tower of Babel 2019

Authoring	Content Model	Local Repo	Remote Repo	Rendering
ArborText	UniDoc	X-Hive	X-Hive	Nokia R16.1 XML CDOC

Product category	Product	Authoring tool/feature
Emerging Products, Security Management	NetGuard Compliance Template Manager NetGuard Audit Compliance Manager Security Workflows Orchestrator	Adobe FrameMaker 12 Unstructured and Oxygen
Digital Experience	SurePay, SPS	ArborText

Authoring	Content Model	Local Repo	Remote Repo	Rendering
Oxygen	DocBook	Subversion	Subversion	Maven, Ant, Jenkins

Digital Network	SBC, DDE, AAA, MSW	ArborText	UniDoc	X-Hive	X-Hive	X-Hive	N/A	N/A	N/A	OLOS
Network Management	OOH	ArborText	UniDoc	X-Hive	X-Hive	X-Hive	N/A	N/A	N/A	OLOS
Digital Intelligence	Analytics Platform	Oxygen	DITA	Git	Gerrit	DITA-OT, Ant, Jenkins	Susagor	Confidence and Artifactory until now for the draft	N/A	OLOS
Digital Intelligence	CA4MH	Oxygen	DITA	Git	Gerrit	DITA-OT, Ant, Jenkins	N/A	non-custom Eclipse Information Center (IC) for SW-Integrated	SW-Integrated IC	NOLS
CloudBand	CBM1									
CloudBand	CBM2									
Network Software	CSF									
Digital Intelligence	Traffice									
CloudBand	CBND									
Digital Intelligence	Customer Experience Management on Demand									
Digital Operator	Performance Manager (MPM)									
Emerging Products	Network Advanced Configurator AC Lite	Oxygen	DITA	SVN	SVN	DITA-OT, Ant, Jenkins	N/A	non-custom Eclipse Information Center (IC) for SW-Integrated delivery and non-custom Eclipse Information Browser (IB) for NOLS	SW-Integrated IC	NOLS
Network Management	NetAct	Oxygen	DITA	SVN	SVN	Jenkins + Maven + Ant	N/A	non-custom Eclipse Information Center (IC) for SW-Integrated delivery and non-custom Eclipse Information Browser (IB) for NOLS	SW-Integrated IC	NOLS
Digital Operator	SON	Oxygen	DITA	SVN	SVN	PDF	N/A	N/A	N/A	NOLS
Emerging Products	Eco-Net	Oxygen	DITA	SVN	SVN	PDF	N/A	N/A	N/A	NOLS
Emerging Products	Scene Analyzer	Oxygen	DITA	SVN	SVN	PDF	N/A	N/A	N/A	NOLS
Digital Experience	IMPACT Data Collector (DDC) /	Oxygen	DITA	SVN	SVN	DITA-OT, Ant, Jenkins	N/A	InfoBrewer, Eclipse InfoCenter	N/A	NOLS

Authoring	Content Model	Local Repo	Remote Repo	Rendering
Oxygen	NSN DITA 1.2	Git	Gerrit	NSN DITA-OT 1.6, Ant, Jenkins

Authoring	Content Model	Local Repo	Remote Repo	Rendering
FrameMaker	Unstructured and Structured	Mercurial	Mercurial	Manual

Digital Intelligence	Fastmail (incl. RTD, Advanced Analytics, GeoIP, Big Data Search, Smart Monitoring, RTP?)	Structure of FrameMaker, Word	DITA	Mercurial	Mercurial repository	manual rendering	N/A	none	SW-Integrated IC	NOLS
Digital Operator	Physical Inventory	Structure of FrameMaker, RakeHelp, Word	DITA	Mercurial	Mercurial repository	Apache Salt	N/A	None	Rakehelp WebHelp for context help in product GUI	NOLS

CNS Harmonized Platform

Front-End

Centralized Back-End

oXygenXML

DITA 1.3 XML
Content
Model

Maven, Ant,
DITA Open
Toolkit

Git Local
Command
Line
TortoiseGit
Oxygen
Plugin

Git Server
(Gerrit)

Kubernetes
Jenkins

Dockerized
Maven, Ant,
DITA Open
Toolkit

Artifactory

Authoring

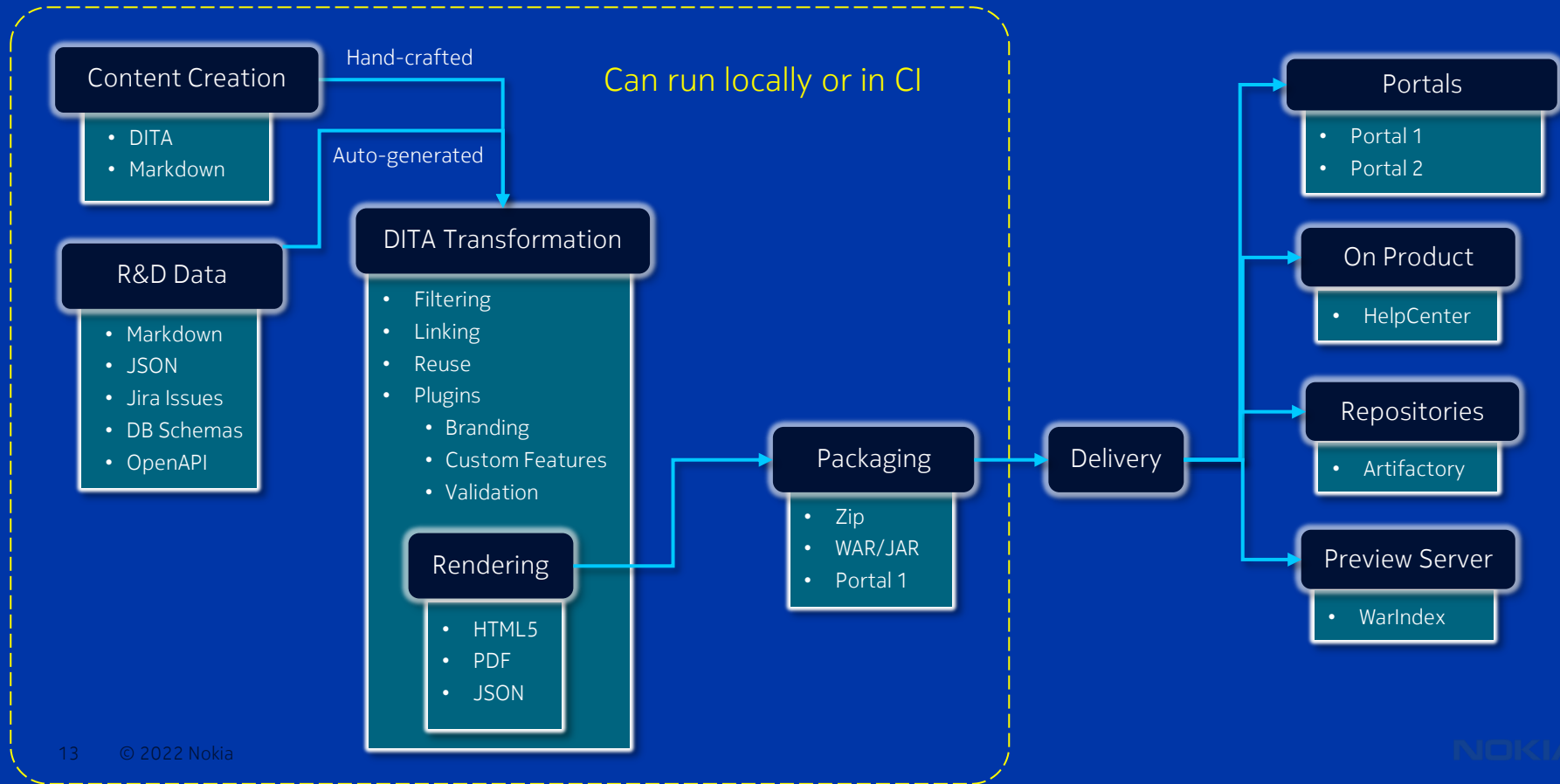
Local
Publishing

Version Control

Server Publishing

Binary
Storage

Content Authoring and Publishing

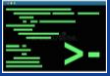


Writer Workstation

Windows Workstation



OxygenAuthor



GitBash



TortoiseGit



Maven



Ant



Python

• Plugins

- Local Builds
- Log Viewer
- Utilities
- DevOps Menu
- Help Menu

• Framework Extensions

- DITA Topics
 - Schematron Rules
- DITA Maps
 - Schematron Rules
 - Refactoring Operations
- Maven
- Ant

• Schematron Rules



• Plugins

- Terminology Checker
 - XML Rules
 - Vale Rules *
- Batch Converter
- DITA References
- Git Client

• Frameworks used

- DITA Topics
- DITA Maps
- Maven
- Ant

Syncrosoft Add-Ons

Custom Oxygen Add-On Build and Deployment

	Name	Version	Author	<oxygen/> Ver...	Type	Status
<input type="checkbox"/>	Batch Documents Converter	5.0.1	Syncro Soft	22.1+	Plugin	Installed
<input type="checkbox"/>	CNSUA Oxygen Framework UAT	3.0.374	Nokia	24.1+	Framework	Installed
<input type="checkbox"/>	CNSUA Oxygen Plugin UAT	2.0.374	Nokia	24.1+	Plugin	Installed
<input type="checkbox"/>	CNSUA XML Validation Rules REVIEW	3.1.290	Nokia	23.1+	Framework	Installed
<input type="checkbox"/>	DITA References View	2.0.1	Syncro Soft	19.1+	Plugin	Installed
<input type="checkbox"/>	Git Client	5.0.1	Syncro Soft	25.0+	Plugin	Installed
<input type="checkbox"/>	OpenAPI Documentation Generator	2.0.0	Syncro Soft	25.0+	Plugin	Installed
<input type="checkbox"/>	OpenAPI Tester	1.2.0	Syncro Soft	25.0+	Plugin	Installed
<input type="checkbox"/>	Terminology Checker	4.1.0	Syncro Soft	22.0+	Plugin	Installed
<input type="checkbox"/>	Vale Validation	1.0.0	Syncro Soft	20.0+	Plugin	Installed
<input type="checkbox"/>	XSpec Framework (required by the XSpec He.2.2.6		Syncro Soft	20.0+	Framework	Installed
<input type="checkbox"/>	XSpec Helper view	2.2.6	Syncro Soft	20.0+	Plugin	Installed

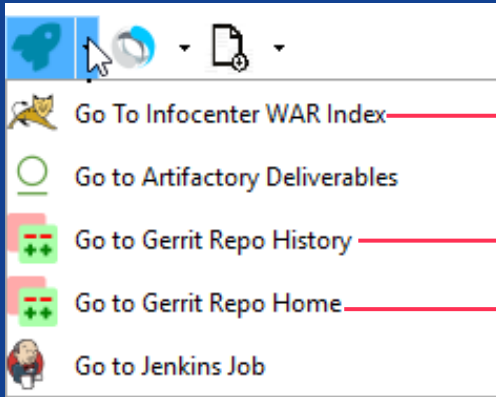
Local CI build compatibility and integration

The image shows a sequence of three windows from an IDE, illustrating the process of configuring a local CI build. Red arrows indicate the flow of the process:

- Build Options (Left):** Shows the 'Output Format' set to 'PDF' and the 'Use advanced options' checkbox unchecked. The 'Build' button is highlighted.
- Input (Middle):** A dialog titled 'Choose a target:' with a dropdown menu showing options: 'main', 'xhtml-pipe', 'xp', 'pdf-pipe', 'pp', 'ed-pipe', and 'main'. The 'main' option is selected.
- Build Options (Right):** Shows the 'Use advanced options' checkbox checked. The 'Build' button is highlighted.

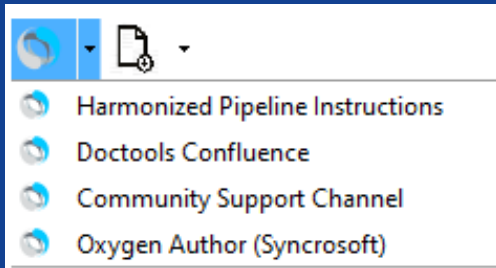
A yellow callout box at the bottom right contains the text: "executes any target with @extensionOf="build", which is what's done by default in a CI build".

Local CI build compatibility and integration



<servers>

```
<service id="tomcat">  
  <hostname>http://yada.yada.nokia.yada</hostname>  
  <page id="infocenter">  
    <pathStart>warindex/?productname-column=</pathStart>  
    <pathEnd>,contains&sorting=date-column,desc</pathEnd>  
  </page>  
</service>
```

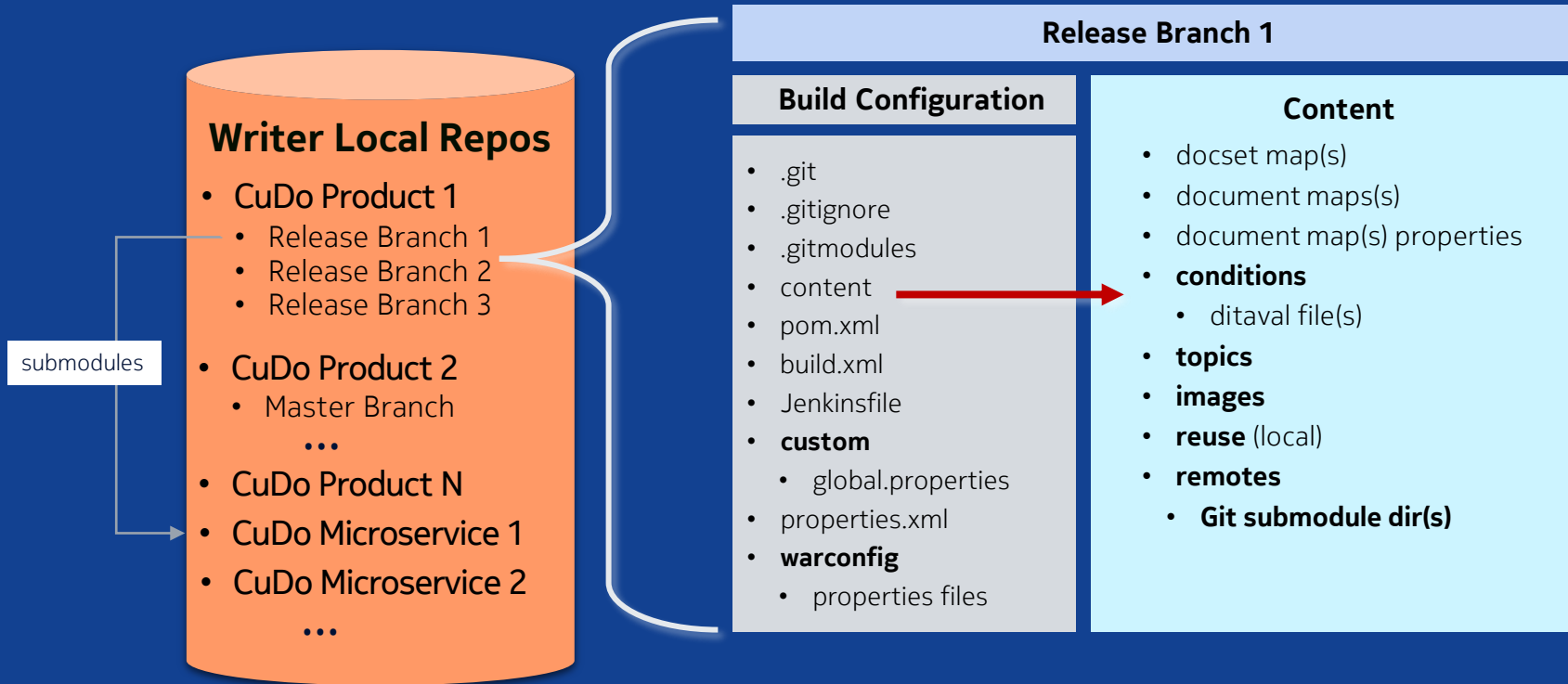


```
<service id="gerrit">  
  <hostname>https://yada.yada.nokia.yada</hostname>  
  <page id="history">  
    <pathStart>gerrit/plugins/gitiles/prefix</pathStart>  
  </page>  
  <page id="project-home">  
    <pathStart>gerrit/admin/repos/prefix</pathStart>  
  </page>  
</service>
```

</servers>

Writer Local Environment

Git Repos, Branching Strategies and Project Layout



Predates OT Project files

<build-doc>

Git Branch N
Build Configuration
<ul style="list-style-type: none">• .git• .gitignore• .gitmodules• pom.xml• build.xml• Jenkinsfile• custom<ul style="list-style-type: none">• global.properties• properties.xml• warconfig<ul style="list-style-type: none">• properties files

- **build.xml**
 - ant target(s)
 - macro call(s)
 - **<build-doc>**

Attributes/parameters	Description
mappath	Required
transtype	Required
docname	File or dir name (defaults to mappath's basename w/o extension)
ditavalfile	One or more if present.
propertyfile	.properties file to apply. Defaults to <mappath basename>.properties.
publish	Defaults to false
launch	Defaults to true
enabled	Defaults to true
security	Defaults to external. Can also be internal or reviewer.
draft-mode	Defaults to no
processing-mode	Defaults to strict
arbitrary-argument	Any name=value pair to pass down

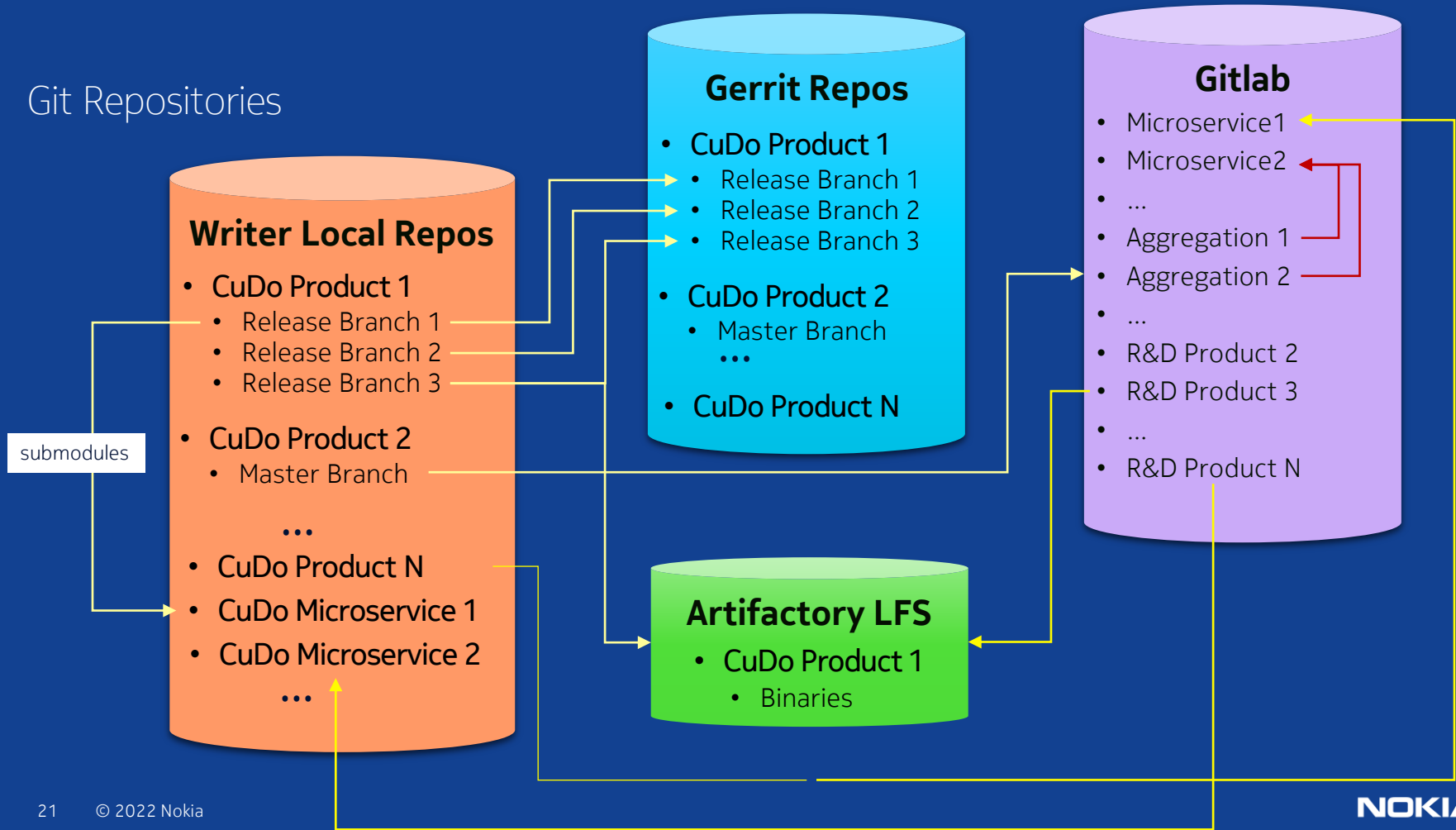
<build-matching-docs>

Git Branch N
Build Configuration
<ul style="list-style-type: none">• .git• .gitignore• .gitmodules• pom.xml• build.xml• Jenkinsfile• custom<ul style="list-style-type: none">• global.properties• properties.xml• warconfig<ul style="list-style-type: none">• properties files

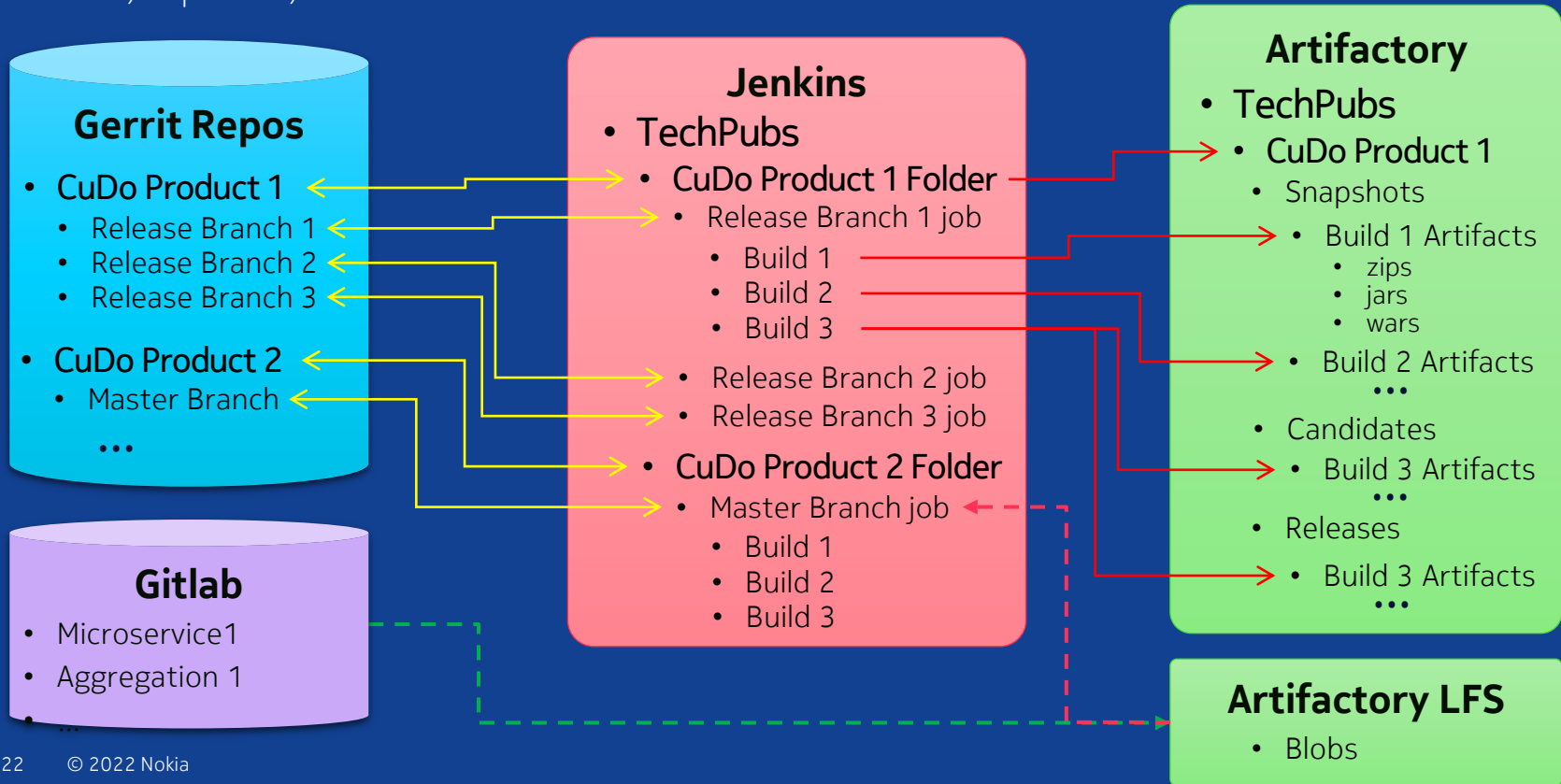
- **build.xml**
 - ant target(s)
 - macro call(s)
 - **<build-matching-docs>**

Attributes/parameters	Description
same as build-doc	plus
dir	directory in which to search for maps
includes	comma-separated list of filename wildcard patterns (ant filesets)
excludes	comma-separated list of filename wildcard patterns (ant filesets)
casesensitive	filename case sensitivity while matching
suffix	a string to add to the end to distinguish multiple matched builds of the same ditamaps (for example, if ditavals used)

Git Repositories



Source, Pipeline, Artifacts



Branch of content to pull during Jenkins build

Single-branch

Multi-branch

Any-branch

```
1 library 'CNSUAHarmonizedPipeline'  
2  
3 def customPipelineParameters = [  
4     myRepository: "my/repository"  
5     myBranch: "myBranchName"  
6 ]
```

```
myRepository: "my/repository"  
myBranch: "multibranch"
```

```
myRepository: "my/repository"  
myBranch: "**"
```

```
9 CNSUA_harmonized_pipeline(customPipelineParameters)
```

Jenkins Shared Library – Basic Parameters

Parameter name	Parameter definition
myRepository (required)	Server Repo path: examples: "NSWTC/abc", "NSWTC/def", "NSWTC/acme". Required for Gerrit Trigger to work properly
myBranch	Branch name: examples: "master", "20.9", "21.3-SP2" <ul style="list-style-type: none">• multi-branch pipelines ignore myBranch whether it's set or not• single-branch pipelines require myBranch for trigger configuration (required)• repo-based pipelines don't have myBranch or have myBranch set to <code>***</code>
notificationList	a comma separated email list for build status notifications. These will be sent when a job fails or when a job succeeds after having previously failed.
useGerritReviews	defines whether the job is triggered by Gerrit review events (true) or push events on the branch (default: false)
manualTriggerOnly	disables automatic job triggers (default: false)
enableArtifactoryNotification	if enabled, successful builds send a notification mail with the link to Artifactory to the user who triggered the build, defaults to true

Jenkins Shared Library – Intermediate Parameters

Parameter name	Parameter definition
mavenBuildGoal	provides a custom maven goal for the pipeline (defaults to deploy)
mavenPom	relative path to the custom POM file to be used in the build
customPreBuildExtension	a function that is triggered before stage "Build"
customPostBuildExtension	a function that is triggered after stage "Build"
customPostAlwaysExtension	a function that is triggered after all builds
customPostSuccessExtension	a function that is triggered on successful builds

Jenkins Shared Library – Advanced Parameters

Parameter name	Parameter definition
dockerRepo	the docker repository name in Artifactory
dockerImage	the docker image ID in Artifactory
dockerTag	the docker image tag in Artifactory
customTriggerProjects	overrides our default gerritProjects value in triggers
customSubmoduleCheckout	overrides the default script in stage 'Submodule Checkout'
useDockerPodTemplate	if true, the job uses 'k8s-dind-rootless', otherwise 'k8s-build'
enablePipelineVersionCheck	if false, disables the version check step in the pipeline so that experiments can be conducted with unreleased or unstable pipeline versions. not recommended for publications
customBuildStep	overrides the script run in pipeline stage "Build"





Schematron Validation as Junit results in Jenkins

All Failed Tests

Test Name	Duration	Age
schematron.warn.topics/201708-support-MS-Exchange-2016-mails.html.dita.bold_not_allowed_in_title./concept[1]/conbody[1]/p[5]/fig[1]/title[1]/b[1]		
Error Details		
Bold element is not allowed in title.	0 ms	2
schematron.warn.topics/201708-support-MS-Exchange-2016-mails.html.dita.img_in_title./concept[1]/conbody[1]/p[5]/fig[1]/title[1]/image[1]		
Error Details		
Image should not be inside the title element of a figure, but after the title element.	0 ms	2

Jenkins pushes built artifacts to Artifactory

43 Artifacts

Artifact Name ^	Type	Repo Path	
docset-22.9-18-docsetmap-resourceid.json	json	techpubs-mvn-releases/com/nokia/...-docset/22.9-18	docset-22.9-18-docsetmap...  View
docset-22.9-18-docsetmap_HTML.zip	zip	techpubs-mvn-releases/com/nokia/...-docset/22.9-18	docset-22.9-18-docsetmap...  Copy url
docset-22.9-18-docsetmap_PDF.zip	zip	techpubs-mvn-releases/com/nokia/...-docset/22.9-18	docset-22.9-18-docsetmap...  Download
docset-22.9-18-..._22.9.war	war	techpubs-mvn-releases/com/nokia/...-docset/22.9-18	docset-22.9-18-..._22.9.war  Show In Tree
docset-22.9-18-..._22.9gui.war	war	techpubs-mvn-releases/com/nokia/...-docset/22.9-18	docset-22.9-18-..._22.9gu...

Docker image cnsua-pipeline

Maven Layer

```
docker run -rm -v ${PWD}:/workspace \  
  internal-artifactory-server.nokia.net/cnsua-pipeline:latest \  
  mvn deploy [-Dtarget=variant1-pdf-internal] \  
  [-Dant.debug=true]
```

- optional other Maven parameters

- Maven command with optional Ant target

- image name

Docker image `cnsua-main-build`

Ant Layer

```
docker run -rm -v ${PWD}:/workspace \  
  internal-artifactory-server.nokia.net/cnsua-main-build:latest \  
  [feature1_platform1] \  
  [-Dant.debug=true]
```

optional Ant target



image name

optional other Ant parameters

Docker image cnsua-pipeline

Ant targets and build-doc macros

```
<target name="platform2_product1_customer5 " extensionOf="build">
  <build-doc
    transtype="cnsua-pdf"
    mappath="content/ platform2_product1_customer5_Operating_Documentation.ditamap"
    ditavalfile="content/conditions/pdf.ditaval, content/conditions/platform2.ditaval,
                content/conditions/product1.ditaval, content/conditions/customer5.ditaval"
    propertyfile="custom/platform2_product1_customer5_Operating_Documentation.properties "
  />

  <build-doc
    transtype="cnsua-eclipsehelp"
    mappath="content/ platform2_product1_customer5_Operating_Documentation.ditamap"
    ditavalfile="content/conditions/html.ditaval, content/conditions/platform2.ditaval,
                content/conditions/product1.ditaval, content/conditions/customer5.ditaval"
    propertyfile="custom/platform2_product1_customer5_Operating_Documentation.properties "
  />
</target>
```


Docker image cnsua-dita-ot

DITA-OT Layer

```
docker run -rm -v ${PWD}:/workspace \  
  internal-artifactory-server.nokia.net/cnsua-dita-ot:latest \  
  --input=/workspace/content/Nokia_feature_variant_Operating_Documentation.ditamap \  
  --format=cnsua-pdf \  
  --filter="content/conditions/html.ditaval: \  
    content/conditions/product1.ditaval: \  
    content/conditions/platform3.ditaval: \  
    content/conditions/cust1.ditaval:" \  
  --output="/workspace/myoutput-directory" \  
  --parallel=true
```

image name

output format

composable
conditional filter
rules

output directory

optional other DITA-OT parameters

UA Rendering Pipeline Refactoring and other Opportunities

- **Single-source metadata** for releases and product variants
- Generation of config files based on **higher-level representation**
- **Actionable Log File Messages** and filtering
- Refactor to **leverage advancements in FOSS components** (DITA-OT Projects)
- **Materially engage with DITA-OT and LwDITA FOSS projects** to strengthen the platforms
 - DITA-OT 4.0
 - Lightweight-DITA (Markdown to DITA build-time transformation)
 - Contribute features, bug fixes, test cases

Custom DITA-OT Plugins (not all)

Name	Description
<code>context.sensitive.help</code>	Transforms <code><resourceid></code> elements into a JSON file that target application can use to locate a specific topic by a unique id.
<code>docset</code>	A type of DITA map that aggregates discrete documents for a product into a single set, but separate documents
<code>docset.resources</code>	Implements cross-deliverable links using peer maps and keys
<code>document.list</code>	Generates a DITA topic that contains a high-level list of the discrete documents in a docset along with metadata
<code>fop</code>	Allows us to update the FOP version independent of the DITA-OT release with a Maven dependency declaration
<code>html</code>	Customizations to the base transform (supports XHTML, HTML5, and EclipseHelp)
<code>pdf</code>	Extension to the OT pdf2 plugin
<code>Schematron</code>	Supports execution of rule checks in the Maven build process. Transforms SVRL to JUNIT format for Jenkins
<code>third-party.copy</code>	Allows writers to reference a zip file in a topicref so that it's included as is in the deliverable package with a TOC entry generated for it
<code>trademarks</code>	Harvests all trademark elements, looks up the usage description, and injects it into the Legal page

Context.sensitive.help

<resourceid> elements serialized as JSON and added to deliverable artifact

```
[{ "appid": "rates",  
  "appname": "yada",  
  "ux-context-string": "page",  
  "ux-windowref": "",  
  "href": "yada_charging_user_guide\/yada_charging_user_guide\/rates.html"  
}, {  
  "appid": "tiered_allowances",  
  "appname": "yada",  
  "ux-context-string": "page",  
  "ux-windowref": "",  
  "href": "yada_charging_user_guide\/yada_charging_user_guide\/chargingstep.html"  
}, {  
  "appid": "threshold_profiles",  
  "appname": "yada",  
  "ux-context-string": "page",  
  "ux-windowref": "",  
  "href": "yada_charging_user_guide\/yada_charging_user_guide\/threshold1.html"  
}]
```

Key benefits: Dockerized Render Engine for Product CI

Current capabilities

- Ingest source content from **DITA or Markdown** files
- Apply **conditional filtering** to produce targeted UA deliverables
- Run **content validation** rules
- Render in latest **HTML and PDF** formats
- **Deploy** to Artifactory and Customer Portal

Enables R&D Pipelines

- **Auto-generate** and insert UA content from R&D sources pre doc-build
- Further refine UA build conditions to **match product variants** for different cloud environments
- Trigger **UA QA checks** in test phase of software build
- Push UA deliverables to desired, **non-default destinations**

TechComm Web Server

- Apache HTTP and Tomcat services
- EclipseHelp WAR hosting from CI builds
- Oxygen Add-Ons hosting
 - CI Builds create updatesite.xml and zips for
 - every Jira issue being worked on
 - every release candidate
 - every release
- WarIndex front-end and back-end (Mika)
 - front end created using same React-based UX components our product devs use so we could expand our skill set into the future
 - back-end REST API for maintaining the list of all WARs and their publication metadata

TechComm Web Server

WarIndex – React launcher for EclipseHelp WAR files

NOKIA CNS Documentation Sets

Category	Product	Release	Date (last modified)	Link	Actions
Analytics	Contains	22.5	2022-08-08 14:59:04	Open	
Miscellaneous	Product 1	FP3v2	2022-08-08 10:15:32	Open	
Miscellaneous	<input type="radio"/> AND <input checked="" type="radio"/> OR	master	2022-08-05 16:22:35	Open	
Test Files	Contains	6.3.3	2022-08-04 00:39:08	Open	
Test Files	Product 2	6.3.3	2022-08-04 00:32:54	Open	
Instructions	dt-toolsinstructions	6.3.3	2022-08-04 00:30:00	Open	
Miscellaneous	Product 7	23	2022-07-27 13:59:48	Open	
Miscellaneous	Product 54	22_MP1	2022-07-26 16:21:53	Open	Copy Link
Miscellaneous	Product 14	master	2022-07-22 07:38:05	Open	Copy Link
Test Files	vsg-pdf	6.3.2	2022-07-21 20:35:43	Open	Copy Link
Test Files	nokia-dita-test	6.3.2	2022-07-21 20:30:50	Open	Copy Link
Instructions	dt-toolsinstructions	6.3.2	2022-07-21 20:27:49	Open	Copy Link

Actions menu items: Clear Sorting, Clear Filters, Copy Link to Filtered Content, Hide Doctools Content, Show Coachmarks, Size Columns to Fit, Reset all Columns, Export to CSV

And then along came Markdown...

- R&D Internal docs were originally in Confluence
- Migrated to Markdown and MkDocs static site generator
- Organic desire to "reuse" Markdown R&D content in customer-facing documentation... somehow

And then along came Markdown...

- Created by John Gruber in 2004 as a simpler way to author blog content than raw HTML (prose)
- No standards-body backed specification (OASIS, W3C, ISO, etc.)
- Flavors
 - CommonMark v0.30 (2021-06-19)
 - Extramark
 - Github Flavored Markdown (GFM)
 - Extensions that are inconsistent across different implementation programming languages
 - Gitlab Flavored Markdown
 - Kramdown
 - MultiMarkdown
 - Pandoc
 - etc

And then along came Markdown...

DITA is to Markdown as Raw Photo Format is to JPEG

Raw Photo Format	JPG
"Exposed" yet not "developed"	Developed Photo
Metadata superset (camera sensor)	Metadata subset: EXIF
Raw camera sensor image data (sub-pixel color values)	Materialized pixels final resolution
Uncompressed - Lossless	Compressed - Lossy
Better detail and dynamic range – more pixel information	Can't get back to rich/raw data. Limited options for superficially tweaking brightness, contrast, white balance, color saturation, and tonal range.
Single raw image data can be retargeted to produce many possible outputs via processing (transformation)	Cannot add detail without inexact interpolation . Above properties can be shifted, but not reproduced at high fidelity.

And then along came Markdown...

JPEG Metadata via EXIF standard

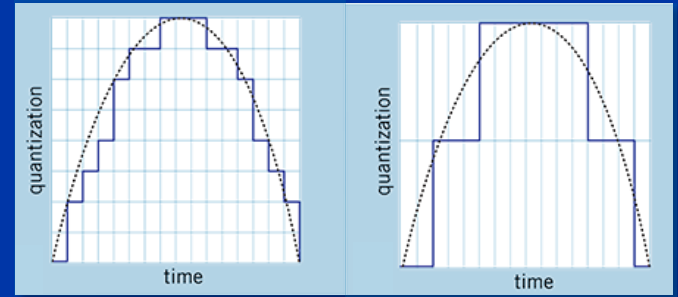
Tag	Value
Manufacturer	CASIO
Model	QV-4000
Orientation (rotation)	top-left [8 possible values ^[27]]
Software	Ver1.01
Date and time	2003:08:11 16:45:32
YCbCr positioning	centered
Compression	JPEG compression
X resolution	72.00
Y resolution	72.00
Resolution unit	Inch
Exposure time	1/659 s
F-number	f/4.0
Exposure program	Normal program
Exif version	Exif version 2.1
Date and time (original)	2003:08:11 16:45:32
Date and time (digitized)	2003:08:11 16:45:32
Components configuration	Y Cb Cr –

Compressed bits per pixel	4.01
Exposure bias	0.0
Max. aperture value	2.00
Metering mode	Pattern
Flash	Flash did not fire
Focal length	20.1 mm
MakerNote	432 bytes unknown data
FlashPix version	FlashPix version 1.0
Color space	sRGB
Pixel X dimension	2240
Pixel Y dimension	1680
File source	DSC
Interoperability index	R98
Interoperability version	(null)

And then along came Markdown...

Other Analogies

- Digital Audio
 - Sample Rate – snapshots of amplitude per second
 - Bit-Depth – The precision of each sample
 - Recording Studio: 192K, 24-bit
 - CD: 44.1K, 16-bit



- ✓ more samples per second
- ✓ more bits to express the amplitude of each sample
- ✓ provides more information that can be leveraged to produce a downstream deliverable

DITA to Markdown: **Downsample** to produce lower-res, downstream, static rendition

Markdown to DITA: **Upsampling** requires inexact interpolation to estimate missing information in low-res source to fill in hi-res gaps

Developer Perspective

Agenda

- Maven Modules
 - pipeline (plugins, testdocs, toolsinstructions)
 - oxygen extension modules (plugin, frameworks, Schematron)
- Gerrit reviews (pull requests)
- Unit Testing Frameworks
 - XSpec (XSLT, Schematron)
 - AntUnit
 - Mockito (Java)
 - Jest (JavaScript)
 - Cypress (JavaScript)
 - Robot (Python)

Automatically triggers builds of each other as is useful. Including the Visual Style Guide content.

NOKIA