



adaptTo()

EUROPE'S LEADING AEM DEVELOPER CONFERENCE  
21<sup>st</sup> – 23<sup>rd</sup> OCTOBER 2024

# Indexing in AEMaaCS- Behind the Scenes

Mohit Kataria, Adobe | Nitin Gupta, Adobe

# About Us



Mohit Kataria

Software Developer, Adobe

Jackrabbit Oak Committer

Nitin Gupta

Software Developer, Adobe

Jackrabbit Oak Committer



- AEM 6.5 vs AEMaaCS
- Oak Pal Rules
- Indexing Job
- Failed Deployment Pipelines
- Key Takeaways



adaptTo()

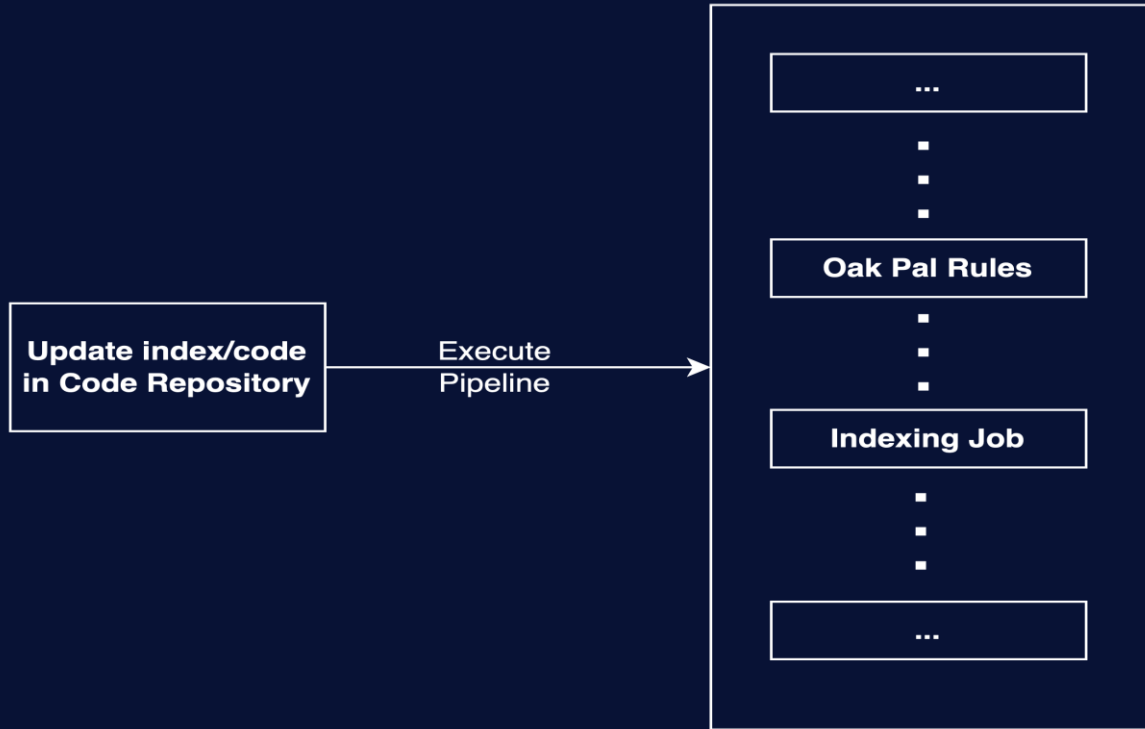
# AEM 6.5 VS AEMaaCS

# AEM 6.5 VS AEMaaS

AEM6.5	AEMaaS
Dedicated Servers	Kubernetes Deployment
Single nodestore	Composite nodestore
Upgrades are performed in maintenance window	Hot upgrade
Update original index	Versioned indexes
Either old or new index works	Old index for old code and new index for new code
Oak performs text extraction.	Another service to extract text.
Can create and update property indexes	Can't create and update property indexes

- Online indexing
  - Directly add/update index definition on AEM
  - Add `reindex=true` to start indexing process
- Out of band indexing
  - Use oak-run utility to update and reindex outside AEM

# Index Customization on AEMaaCS





adaptTo()

# Oak Pal Rules



- Static analysis of index definition.
  - NodeType of Index must be oak:IndexDefinition
  - Tika config in DamAssetLucene should only extract from text files.
  - Complete list in appendix.



adaptTo()

# Indexing Job

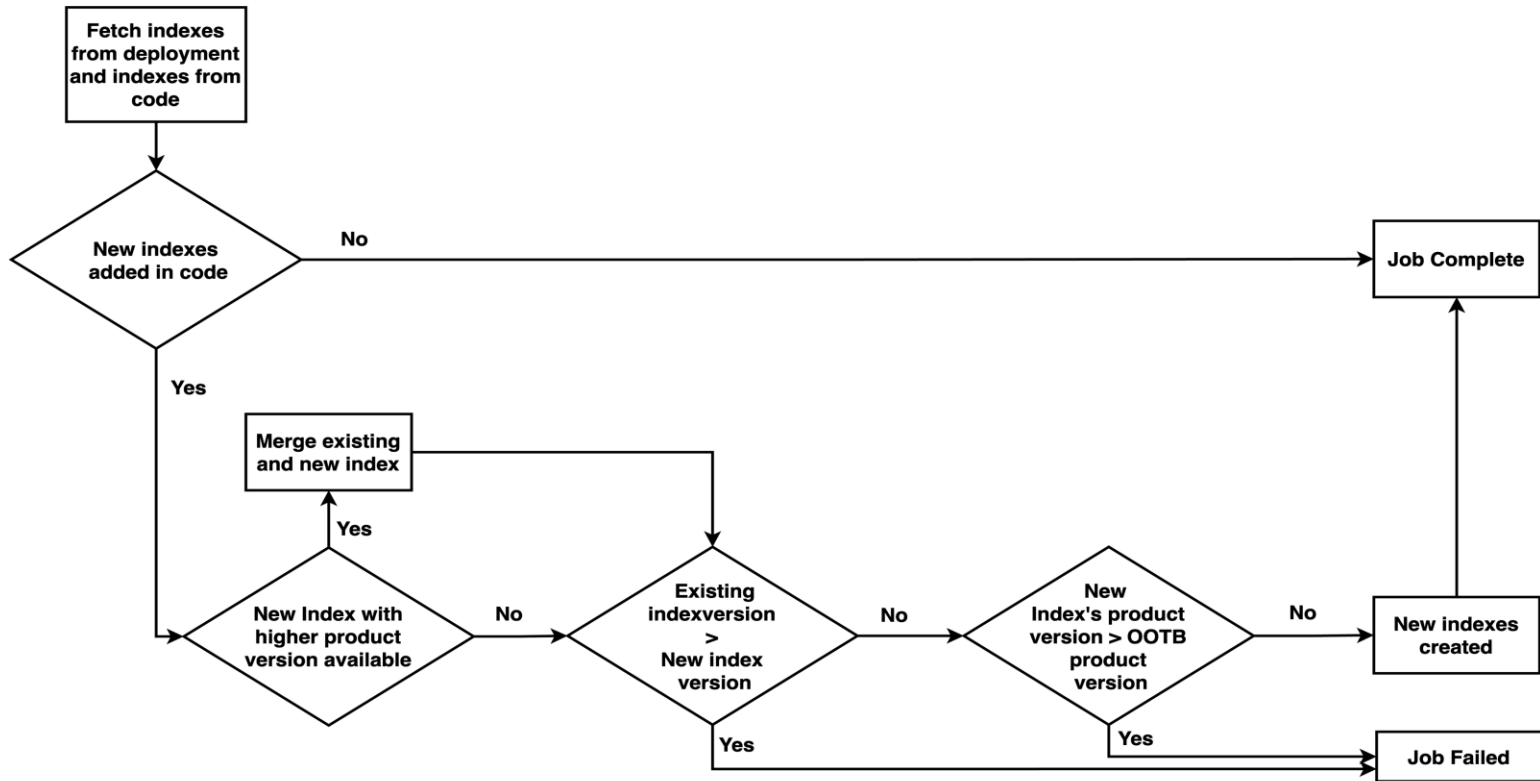
# Index Naming Convention & Versioning

- damAsset-8-custom-10
  - Base Name = damAsset
  - Product Version = 8
  - Customer Version = 10
  
- \* custom is reserved word

# Index Comparison

- `damAsset-x-custom-y > damAsset-p-custom-q`
  - Comparison is valid for indexes with same Base Name
  - $x > p$
  - $x=p \ \&\& \ y>q$

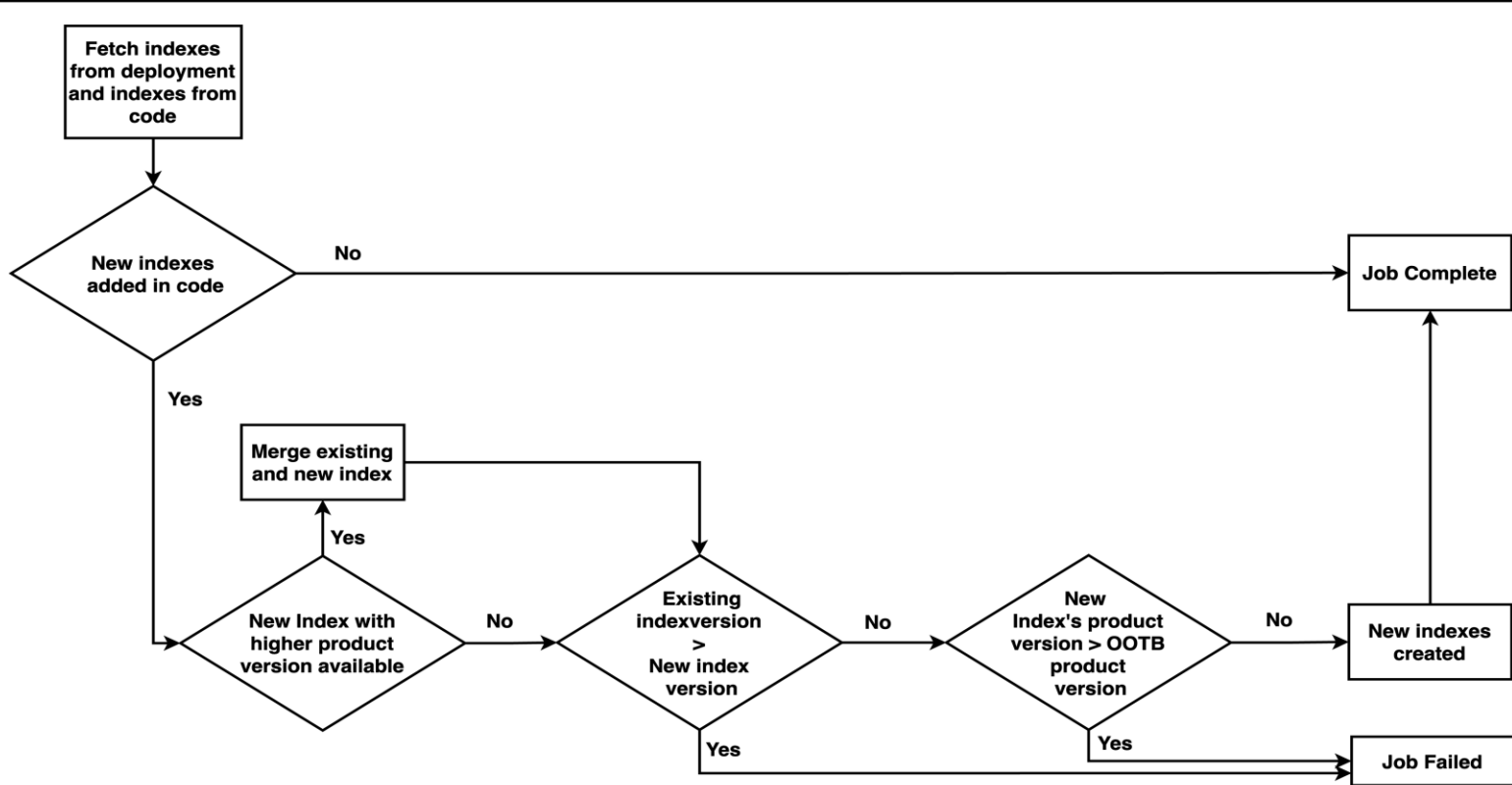
# The Indexing Job



# Index Merging Example

- Existing Index: damAssetLucene-8-custom-9
- New Index: damAssetLucene-9
- 3-way Merging:
  - damAssetLucene-8
  - damAssetLucene-8-custom-9
  - damAssetLucene-9
- Merged Index: damAssetLucene-9-custom-1

# The Indexing Job



# New Index Version Constraints

- Existing Index: damAssetLucene-8-custom-9
  - Valid index names:
    - damAssetLucene-8-custom-10
    - damAssetLucene-8-custom-12
  - Invalid index names:
    - damAssetLucene-7-custom-15
    - damAssetLucene-8-custom-7
    - damAssetLucene-9-custom-2

Note: These are just index name examples, we don't create multiple custom indexes with same base name.





adaptTo()

# Failed Deployment Pipelines

# Deploy Step Logs

- Can not add a new customized index `damAssetLucene-10-custom-2` because an index (possibly merged) named `damAssetLucene-11-custom-1` that has a higher major version already exists. As higher major versions take precedence, the new customized index will not be used. In order to resolve the problem, the new customized index needs to be merged with the latest index, and renamed, so that it has a higher version number than the latest index. Suggested name for new index `damAssetLucene-11-custom-2`
- Cannot add a new customized index `damAssetLucene-11-custom-1` because the `product version for this index is 11` which is higher than the product index of the existing latest index `damAssetLucene-10-custom-1`. In order to resolve the problem, consider creating the new customized version with `product version equal to 10`.

# Key Takeaways

- Index versioning is an important task.
- Be cognizant of the versions available on the system before introducing a new custom index.
- Don't update/remove OOTB index properties.



adaptTo()

# Have a Happy Indexing Experience



adaptTo()

# Appendix

# Oakpal Rules

CustomFulltextIndexesOfTheDamAssetCheck	IndexAsyncCheck	IndexIncludedPathsWithoutQueryPaths
IndexNodeWithDifferentAnalyzedConfigCheck	IndexNotFulltextGenericNodetypes	IndexNameCheck
IndexCompatVersionCheck	IndexNotUnderUIContentCheck	RestrictIndexCustomizationCheck
<b>IndexDamAssetLuceneCheck</b>	IndexReindexAsyncPropertyCheck	IndexTypeCheck
IndexDescendantNodeTypeCheck	IndexReindexPropertyCheck	OakIndexLocationCheck
IndexDetectMultipleActiveVersionsOfSameIndex	IndexRulesNodeCheck	IndexHasValidTagsProperty
IndexFullyCustomNameCheck	IndexSeedPropertyCheck	IndexTikaConfigCheck