

**Oak Index Migration from AMS to AEM as a Cloud Service: A Comprehensive Guide**

Before diving into the detailed explanation, here's a concise overview of the key concepts covered in this guide:

**Executive Summary**

Oak indexes are critical components in AEM that optimize search performance by creating specific pathways to content. When migrating from AMS to AEMaaCS, index definitions require transformation since AEMaaCS only supports Lucene-type indexes and follows a strict deployment pattern. The Index Converter tool automatically converts existing index definitions to be cloud-compatible, handling the complexities of merging customizations while maintaining search performance. This migration process is essential for ensuring optimal content search capabilities in AEMaaCS while adhering to the platform's immutable/mutable content separation principles.

**Detailed Concept Breakdown**

**Understanding Oak Indexes and Their Role**

Oak indexes are specialized structures in Adobe Experience Manager that enhance search performance within the content repository. Apache Jackrabbit Oak, which powers AEM's content search functionality, uses these indexes to efficiently locate content without traversing the entire repository tree. Without proper indexing, AEM would need to scan through potentially millions of content nodes to fulfill search queries, significantly impacting performance.

In traditional AEM deployments (on-premise or AMS), Oak indexes are stored under the /oak:index node in the repository. They define both the index configuration and store the actual index data in a node structure that's invisible to standard AEM tooling[[1]](#fn1)[[2]](#fn2).

**Key Differences in AEMaaCS**

When moving to AEM as a Cloud Service, several important changes affect how indexes are managed:

**Supported Index Types**

AEMaaCS **only supports Lucene Oak indexes**. If your existing implementation uses other index types like property indexes or Solar indexes, you must convert them to Lucene format[[3]](#fn3).

**Blue-Green Deployment Model**

AEMaaCS uses a blue-green deployment model for indexes, allowing for zero-downtime deployments and fast rollbacks. This means that while a new index version is being built, the old one continues to serve queries[[3]](#fn3).

**Naming Convention**

Index definitions in AEMaaCS follow a versionable naming scheme that always increments:

<indexName>[-<productVersion>]-custom-<customVersion>

For example, customizing the built-in damAssetLucene index would result in damAssetLucene-custom-1[[3]](#fn3)[[4]](#fn4).

**Deployment Process**

Unlike in AMS where you could directly modify indexes in the repository, in AEMaaCS:

* No index manager tool is available
* All index changes must go through the CI/CD pipeline
* Indexes must be part of the ui.apps package
* Sub-directories are not supported in index definitions[[4]](#fn4)

**Content Separation in AEMaaCS**

AEMaaCS implements a strict separation between:

* **Immutable content** (code): /apps and /libs - built into the container image
* **Mutable content**: Everything else including /content, /conf, /var, /etc, /oak:index, etc.

Even though /oak:index is classified as mutable, index definitions must be deployed with immutable code through the Cloud Manager pipeline before any mutable content is deployed[[5]](#fn5).

**The Index Converter Tool**

**Purpose and Functionality**

The Index Converter is a utility developed specifically to migrate existing Custom Oak Index Definitions to be AEM as a Cloud Service compatible. It's designed to eliminate the manual effort involved in transforming index definitions[[6]](#fn6)[[7]](#fn7).

**How It Works**

The conversion process follows these steps:

1. **Parse definitions**: The tool analyzes both custom and out-of-the-box index definitions
2. **Identify customizations**: It identifies the delta (customizations) made to standard indexes
3. **Validate changes**: It checks if these customizations comply with AEMaaCS guidelines
4. **Merge changes**: It combines the validated customizations with corresponding Oak index definitions
5. **Apply naming convention**: It maintains the custom Oak index definitions and adds "custom" at the end with a hyphen and number (e.g., -custom-1)[[7]](#fn7)

**Ways to Use the Index Converter**

You can use the Index Converter through:

1. **Adobe I/O CLI**: Using the aio-cli-plugin-aem-cloud-service-migration plugin:

aio aem-migration:index-converter

1. **Standalone utility**: Running it directly as a Java application[[6]](#fn6)[[8]](#fn8)

**Practical Applications**

**Migration Workflow for Oak Indexes**

When migrating from AMS to AEMaaCS, follow these practical steps:

1. **Analyze existing indexes**: Use the Best Practices Analyzer tool to identify what indexes need migration.
2. **Convert index definitions**: Use the Index Converter tool to transform your existing indexes into AEMaaCS-compatible formats.
3. **Place converted indexes correctly**: Position them in your project structure under ui.apps/src/main/content/jcr\_root - remember that sub-root folders are not supported[[4]](#fn4).
4. **Configure package properties**: Ensure your ui.apps package has the following configuration in your Maven POM file:

<packageType>mixed</packageType>

1. **Deploy through Cloud Manager**: Push your changes through the Cloud Manager CI/CD pipeline to deploy the indexes to your AEMaaCS environment[[9]](#fn9).

**Common Migration Scenarios**

**Custom Index Creation**

When creating entirely new custom indexes for AEMaaCS:

1. Define your Lucene index according to AEMaaCS guidelines
2. Follow the naming convention: <indexName>-custom-1
3. Place it in the correct project structure
4. Deploy through Cloud Manager

**Modifying Existing OOTB Indexes**

When you need to customize existing out-of-the-box indexes:

1. Create a new definition based on the original
2. Apply your customizations
3. Name it properly: <originalIndexName>-custom-1
4. Deploy through Cloud Manager

**Converting Ensure Oak Definitions**

If your AMS implementation uses Ensure Oak Index:

1. Convert these definitions to standard Oak Index Definitions first
2. Then migrate them to AEMaaCS-compatible Custom Oak Index Definitions
3. Apply proper naming and deployment[[1]](#fn1)

**Important Quotes or Data Points**

> "The Index Converter allows AEM developers to migrate existing custom OAK index definitions to AEM as a cloud service compatible custom OAK index definitions."[[7]](#fn7)

> "AEM as a Cloud Service only supports Oak Lucene indexes."[[2]](#fn2)

> "Ensure definitions are not supported on AEM as a cloud service. And hence, they need to be converted to OAK index definitions first and then need to be migrated to custom OAK indexes."[[7]](#fn7)

> "The reindexing of the repository after index definition changes requires time and it depends on the size of the repository."[[2]](#fn2)

> "The only acceptable non-erring conditions for reindexing Oak indexes are if the configuration of an Oak index has changed."[[10]](#fn10)

**Final Key Takeaways**

* **Lucene Only**: AEMaaCS exclusively supports Lucene-type Oak indexes; all other index types must be converted.
* **Naming Convention**: Index definitions must follow the specific pattern <indexName>[-<productVersion>]-custom-<customVersion>.
* **Package Structure**: Custom indexes must be deployed as part of the ui.apps package with packageType set to mixed.
* **Deployment Path**: Place index definitions under ui.apps/src/main/content/jcr\_root - sub-directories are not supported.
* **Tool Assistance**: Use the Index Converter tool to automatically transform existing indexes to be AEMaaCS-compatible.
* **Deployment Model**: Understand the blue-green deployment model that ensures zero downtime during index updates.
* **No Manual Modification**: All index changes must go through the Cloud Manager CI/CD pipeline; there's no index manager available.
* **Content Separation**: Indexes are deployed with immutable content but considered part of the mutable repository.
* **Service User**: Ensure proper permissions for the service user that manages indexes.
* **Reindexing Considerations**: Reindexing should be avoided when possible as it requires significant time depending on repository size.

**Curated Educational Resources**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Title of the Resource | Format | Difficulty | Brief Description | Verified URL |
| Index Converter Documentation | Article | Medium | Official Adobe documentation on the Index Converter tool, including its purpose, functionality, and usage instructions. | <https://experienceleague.adobe.com/en/docs/experience-manager-cloud-service/content/migration-journey/refactoring-tools/index-converter> |
| Indexing Best Practices in AEM | Article | Easy | Comprehensive guide to indexing best practices in AEM, covering out-of-the-box indexes, custom indexes, and performance considerations. | <https://experienceleague.adobe.com/en/docs/experience-manager-learn/foundation/development/understand-indexing-best-practices> |
| Search and Indexing in AEM as a Cloud Service | Video/Article | Medium | Hands-on guide exploring how to define and deploy Oak indexes to AEM as a Cloud Service with practical examples. | <https://experienceleague.adobe.com/en/docs/experience-manager-learn/cloud-service/migration/moving-to-aem-as-a-cloud-service/search-and-indexing> |
| How to Deploy Custom Oak Indexes in AEM as a Cloud Service | Blog | Medium | Step-by-step guide for deploying custom Oak indexes in AEMaaCS, including project configuration and best practices. | <https://www.initialyze.com/insights/how-to-deploy-custom-oak-indexes-in-aem-as-a-cloud-services> |
| AIO CLI Plugin for AEM Cloud Service Migration | GitHub | Hard | Technical documentation for the Adobe I/O CLI plugin that helps with AEM as a Cloud Service code refactoring, including the Index Converter. | <https://github.com/adobe/aio-cli-plugin-aem-cloud-service-migration> |
| Step-by-Step Guide to Migrating Oak Indexes in AEM | Article | Medium | Practical guide with commands and examples for migrating Oak indexes between AEM servers. | <https://www.linkedin.com/pulse/step-by-step-guide-migrating-oak-indexes-aem-divanshu-goyal> |
| Get Ready for the Cloud! - AEM Cloud Service Migration Best Practices | Video | Medium | Comprehensive video presentation covering AEM Cloud Service migration best practices, including content transfer and repository modernization. | <https://www.youtube.com/watch?v=7BnKSQHk13I> |
| Using the Repository Modernizer Tool for a Smooth Migration to AEMaaCS | Blog | Easy | Overview of the Repository Modernizer tool and how it helps with project restructuring when migrating to AEMaaCS. | <https://blogs.perficient.com/2023/10/09/aem-as-a-cloud-service-and-repository-modernizer-tool/> |
| Oak Index Deployment in AEMaaCS Project | Forum | Medium | Community discussion about Oak index migration, project structure, and deployment challenges specific to AEMaaCS. | <https://experienceleaguecommunities.adobe.com/t5/adobe-experience-manager/oak-index-deployment-in-aemaacs-project/m-p/451438> |
| Jackrabbit Oak – Repository Migration | Documentation | Hard | Technical documentation on repository migration tools in Jackrabbit Oak, including tools for copying content between repositories. | <https://jackrabbit.apache.org/oak/docs/migration.html> |

⁂

1. <https://adobe-consulting-services.github.io/acs-aem-commons/features/ensure-oak-index/index.html>

1. <https://experienceleague.adobe.com/en/docs/experience-manager-learn/foundation/development/understand-indexing-best-practices>

1. <https://experienceleague.adobe.com/en/docs/experience-manager-learn/cloud-service/migration/moving-to-aem-as-a-cloud-service/search-and-indexing>

1. <https://experienceleaguecommunities.adobe.com/t5/adobe-experience-manager/oak-index-deployment-in-aemaacs-project/m-p/451438>

1. <https://blogs.perficient.com/2023/10/09/aem-as-a-cloud-service-and-repository-modernizer-tool/>

1. <https://experienceleague.adobe.com/en/docs/experience-manager-cloud-service/content/migration-journey/refactoring-tools/index-converter>

1. <https://experienceleague.adobe.com/en/docs/experience-manager-learn/cloud-service/migration/cloud-acceleration-manager/index-converter>

1. <https://github.com/adobe/aio-cli-plugin-aem-cloud-service-migration>

1. <https://www.initialyze.com/insights/how-to-deploy-custom-oak-indexes-in-aem-as-a-cloud-services>

1. <https://github.com/AdobeDocs/experience-manager-65.en/blob/main/help/sites-deploying/best-practices-for-queries-and-indexing.md>