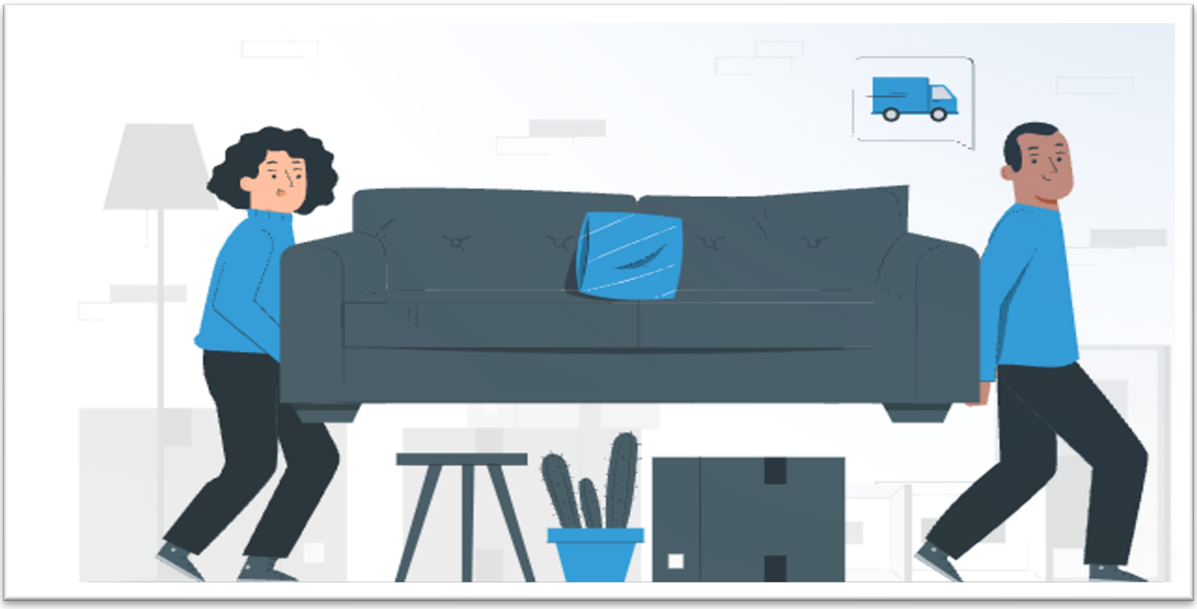
Enterprise to Micro Integrator Migration Best Practice Guide

Product migration is a necessary evil for your Enterprise



Introduction

The context of this article is based on WSO2 Enterprise Integrator (EI) to Micro Integrator (MI) migration. However, the essence of the article can be employed for other WSO2 product migrations such as WSO2 API Manager and WSO2 Identity Server. The intention of this article is to provide you with a comprehensive guideline to structure your migration strategy in the most productive, hassle-free, and cost-effective manner. Whether you're evaluating the benefits of migration, determining the optimal timing, exploring migration options, or preparing for the technical challenges and risks ahead, this guide will provide you with the insights needed to navigate this transformational journey successfully.

Are you a decision maker in your organization? If so, we dedicate this article to make your decision-making process more streamlined when it comes to WSO2 product migrations. Join us as we explore a better integration system based on WSO2 Micro Integrator which is equipped for the long-term sustainability of your organization/Integration ecosystem.

1. Why should you migrate?

Let’s have a look on the flip side of the story, if you do not migrate the product on time. In nutshell, it does massively increase your operational cost and compliance risk by running on outdated components. To be more specific, full product support is available only within 4 years' time duration starting from the product released date (Diagram 1.0). If you are unable to migrate into the newer version of the product end of 4th year, additional arrangements need to be organized with WSO2 by incurring additional cost as well (Extended support).

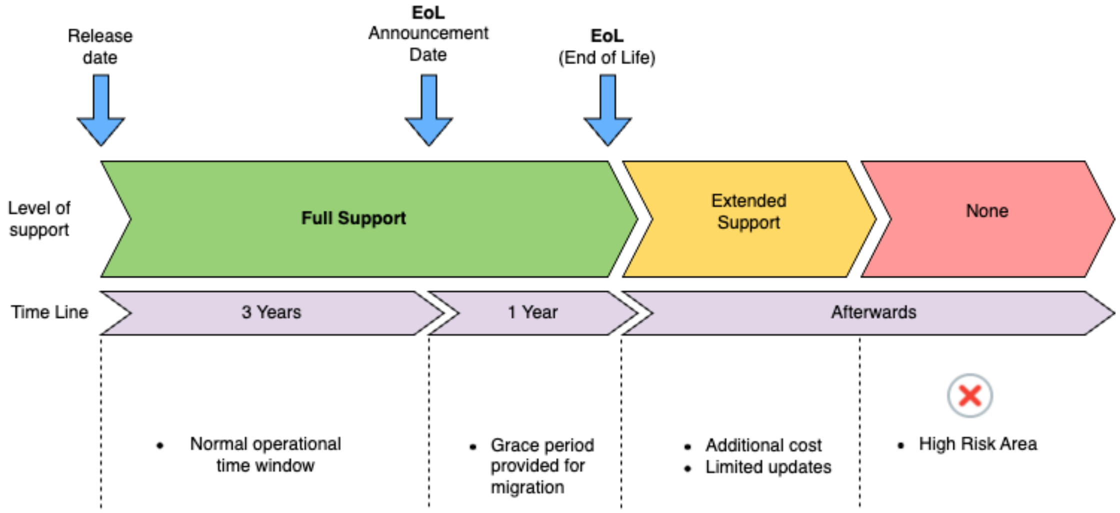


Diagram 1.0

Even though extended support is available with additional cost, availability of security patches and proactive updates are limited to the older versions of the products. On top of that, capability of feature back porting and forward porting is very limited because source code and implementation is quite older compared to the latest version of the product in use.

Another hassle, you might have to face during extended support period is not being able to move forward with cutting edge technologies. For an instance if you are a WSO2 Enterprise Integrator 6.6.0 customer which was released on 22nd Dec 2019, soon will be falling to the category of extended support. At the same time if you are planning to move your integration solution into the container based micro service architecture, having an older version of Enterprise Integrator is a clear bottle neck to update your integration platform.

Finally, even with extended support there is a limit you can go. Most of the product which had been released even before 2017 are now on discontinued state so that you cannot expect any support from WSO2. This is the dangerous zone must consciously look and avoid.

2. When should you migrate?

The answer to the question when you should migrate depends on the current product version you have. Let's have a look on few instances to get proper understanding on timelines.

Product release date	Date to initiate migration	Date to end migration
Enterprise Integrator 6.6.0	2022 Q4	2023 Q4
Enterprise Integrator 7.1.0	2023 Q3	2024 Q3
Micro Integrator 4.0.0	2024 Q2	2025 Q2
Micro Integrator 4.1.0	2025 Q1	2026 Q1

Table 2.0

Referencing to Table 2.0, after 3 years from product released date, WSO2 would announce that within another 1 year of time, given version of the product would reach to the EOL (End of Life) so that support will not be provided after EOL under general support agreement. After 4 years from product released date, you have two choices, either migrate your solution to the latest product version or continue with existing product version under special arrangement made for extended support. The important point is, if you wait till EOL date without doing any pre-planning activities, you might end up with bearing more additional cost than originally anticipated with extended support. This chapter will provide you a guideline to structure your migration plan with respect to product life cycle.

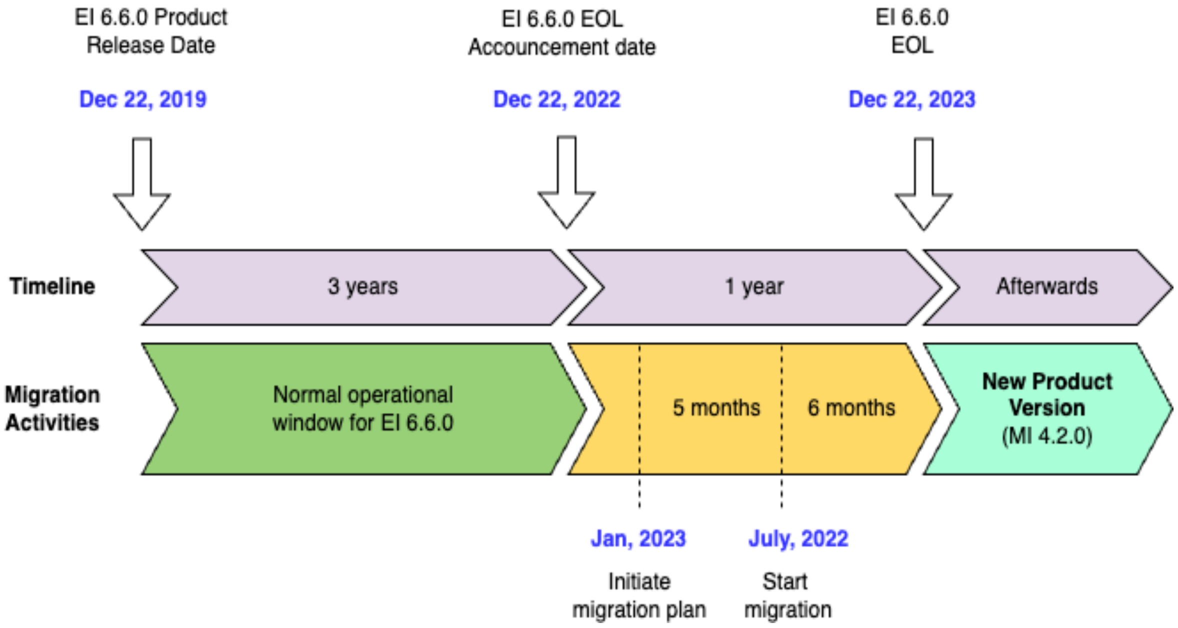


Diagram 3.0

To explain the process with concrete examples, we have taken EI 6.6.0 to MI 4.2.0 migration where EI 6.6.0 comes to EOL by December 2023. As you can see in the Diagram 3.0, EI 6.6.0 was released in December 2019 and EOL announcement was already made in December 2022. In an ideal scenario what you should have done is just after EOL announcement, you initiate a discussion regarding migration strategy for new product version. At the initial stage you might have to take some high-level decisions like which product version needs to be picked up for migration, who is responsible for migration and how to perform migration with minimum impact to other operational activities and stake holders. In the given situation if you initiate the migration at the beginning in 2023, latest release at that time is MI 4.1.0 which is coming close to 1 year old soon. Most prudent decision at that time would be wait few more months and get the next released version (MI 4.2.0) for migration. Once you have all the ingredients in your hand, you can start the actual migration. As a best practice you can start the migration 6 months before EOL date so that you will get enough time to deal with feature gaps, testing of integrations and do necessary performance tests. However, these timelines can slightly vary depending on the complexity of the platform and number of integrations.

3. General release cycle of WSO2 products

According to Table 1.0, you should be able see release cycle for WSO2 Micro Integrator and WSO2 API manager is going on hand in hand. Starting from API Manager 4.0.0 release, each Api Manager release has corresponding Micro Integrator release as well. However, you should be able to observe that after Enterprise Integrator 7.1.0 release, there is no more EI releases. The idea is EI has been discontinued after 7.1.0 release and you must adapt to Micro Integrator to continue with your integration journey. Just in case if you are an EI 6.6.0 user, your general support agreement will come to end on 22nd Dec 2023. The obvious option is you must migrate to latest version of substitute product which is Micro Integrator 4.2.0.

Product	Release Date	End of Life (EOL) Announcement date	(EOL) End of Full Support
Enterprise Integrator 6.6.0	Dec 22, 2019	Dec 22, 2022	Dec 22, 2023
Enterprise Integrator 7.1.0	Aug 12, 2020	Aug 12, 2023	Aug 12, 2024
Micro Integrator 4.0.0	May 05, 2021	May 05, 2024	May 05, 2025
API Manager 4.0.0	May 05, 2021	May 05, 2024	May 05, 2025
Micro Integrator 4.1.0	Apr 11, 2022	Apr 11, 2025	Apr 11, 2026
API Manager 4.1.0	Apr 11, 2022	Apr 11, 2025	Apr 11, 2026
Micro Integrator 4.2.0	Mar 22, 2023	Mar 22, 2026	Mar 22, 2027
API Manager 4.2.0	Mar 22, 2023	Mar 22, 2026	Mar 22, 2027

Table 1.0

4. Distinct options available for migration

Keeping your enterprise integration stack up to date is crucial in today's fast-paced digital landscape. For organizations using the integration platform of WSO2, the need to initiate a migration process may arise for various reasons, such as updates, security enhancements, or changes in business requirements. In this article, we'll explore three approaches to kickstart the migration process.

1. Migration using dedicated Resources from WSO2

One approach is to allocate a dedicated resource from WSO2 itself. This service is usually offered at a premium price but WSO2 possess in-depth knowledge of their products, making them invaluable for complex migrations. But when it comes to customizations, DevOps and end to end testing you will not get much flexibility over this approach.

2. WSO2 Partner Company Collaboration

Another viable option is to engage with a WSO2 partner company. These partners, including Value Added Resellers, OEM Technology Partners, and Distributors, often specialize in WSO2 product implementations. Even though this approach would provide you with the optimum result, the outcome could vary depending on the migration expertise of the partner company.

3. Leverage internal organizational resources

The third option is to leverage internal organizational resources via WSO2 support portal to manage the integration within the organization itself. This approach offers a cost-effective solution for straightforward migrations but comes with the drawback of limited support for customizations and extension points. Also, you may need to build WSO2 expertise with your organization that will consume more time and effort.

Options	Pros	Cons
Dedicated Resources from WSO2	• Faster solutions for product level limitations and bugs due to In-depth knowledge of WSO2 products	• High Cost • Lack of flexibility for customizations, DevOps, etc.
WSO2 Partner Company Collaboration	• Faster solutions to common requirements and known chaos	• Productivity of the migration depends on the resource person

	• Maturity and experience to deal with WSO2 platform • Less cost compared to first option • Possibility of making manage service agreements	• WSO2 platform knowledge will not be transferred to inhouse developers
Leverage internal organizational resources	• WSO2 platform knowledge will be transferred to inhouse developers • Relatively low cost	• Delays can be occurred due to incompetencies of inhouse developers • Knowledge gaps on getting full advantage of customizations & extension points

5. Basic steps of migration

In the case of migrating from WSO2 Enterprise Integrator to Micro Integrator, it's crucial to recognize several pivotal steps that are required to be followed during the migration process. This includes consolidation of configurations into a single file, transition to a file-based registry, clustering via databases, adaptation to schema changes, monitoring capabilities through a dashboard, and adoption of ELK-based analytics. Each of these steps are fundamental pillars of your migration journey.

1. TOML-Based Single Configuration File

Begin your migration journey by embracing TOML-based product configurations, a feature introduced in WSO2 Micro Integrator. With this upgrade, all server-level configurations for your Micro Integrator instance can be conveniently managed using a single configuration file, known as deployment.toml.

To assist you further, refer to the comprehensive list of configuration parameters available for use in the deployment.toml file, each accompanied by detailed descriptions. For in-depth guidance on applying these product configurations to your Micro Integrator deployment, consult the product installation and setup documentation [2].

2. File-Based Registry

As you advance in your migration journey, it's essential to note that WSO2 Micro Integrator comes equipped with a user-friendly file-system-based registry. It comprises of a root registry folder where you'll find sub-registry folders designed to enhance organization and flexibility (Local, Config, Governance).

You have the freedom to store artifacts bundled as carbon applications in any of these folders based on your specific requirements and use cases. This directory structure is thoughtfully maintained to ensure backward compatibility with previous versions of WSO2 Enterprise Integrator.

3. RDBMS-Based Coordination for Clustering

Moving forward with your migration, it's crucial to enable efficient communication among nodes in your Micro Integrator cluster. In contrast to previous versions, WSO2 Micro Integrator relies on RDBMS-based coordination where all nodes collaborate via a database instead of the previous Hazelcast-based clustering mechanism used in WSO2 Enterprise

Integrator. This shift enhances reliability and scalability within your clustered environment, making it essential to provision a database schema to support this new clustering approach.

4. Schema Changes to the User Store

In the migration process, addressing changes in the user store is a pivotal step. By default, the Micro Integrator employs a file-based user store. The move from a file-based user store to LDAP or RDBMS used in your prior version of WSO2 EI may require adjustments in the schema structure to align with the new requirements and functionalities introduced in WSO2 Micro Integrator.

5. MI Dashboard for Monitoring and Management

The MI dashboard represents a modern alternative to the traditional management console found in WSO2 Enterprise Integrator. The Micro Integrator dashboard steps in as a solution to monitor MI instances within your deployment, whether it's a single instance or a group of MI instances forming a cluster. This dashboard offers a dynamic graphical representation of the integration artifacts deployed across your MI instances. More information regarding this can be found here [3].

6. Elastic Stack-Based Operational Analytics

Micro Integrator introduces robust support for publishing operational analytics, seamlessly integrated with the Elastic Stack. This acts as the alternative to the analytics profile offered by WSO2 EI. To enable operational analytics with WSO2 Micro Integrator, you'll need to leverage components from the Elastic Stack, including: Kibana, Elasticsearch, Logstash, & Filebeat. More information regarding this can be found here [4].

6. Technical challenges of migration

The migration from WSO2 Enterprise Integrator to Micro Integrator holds significant promise for enhancing your integration platform. However, it's essential to recognize and address the technical challenges to ensure a smooth transition. In this article, we will dive into these challenges and explore the factors that must be taken into consideration when proceeding with the migration process.

1. Data Migration

The first significant challenge in the migration process revolves around data migration. WSO2 EI relies on a registry database to store artifacts, configurations, and metadata. In contrast, Micro Integrator employs Composite Application (CAR) files for artifact management. Transitioning data from the database to CAR apps requires careful planning and execution. You must ensure data integrity, structure, and consistency throughout this migration.

2. Configuration Migration

Configurations play a pivotal role in the functioning of an integration platform. While Micro Integrator simplifies configuration management through a single TOML file, challenges may arise due to WSO2 documentation gaps. It is vital to ensure that your existing configurations are documented and cross-referenced during the migration process. A configuration catalog of WSO2 MI can be found in [2].

3. Integration-Related Migration

A substantial portion of the migration process involves addressing integration-related challenges. WSO2 EI comes equipped with built-in connectors for various systems and protocols. Initially, ensuring seamless compatibility with the latest or compatible versions of these connectors in Micro Integrator presents a multifaceted challenge. Secondly, if your organization has developed custom integration components or mediators for WSO2 EI, these components may require substantial refactoring to seamlessly operate with Micro Integrator. And finally, the transition to Micro Integrator may necessitate alterations in the package names and namespaces of your custom components and integrations to align with the latest Micro Integrator versions.

7. Risks associated with the migration

Migrating from WSO2 Enterprise Integrator to Micro Integrator is a pivotal process for enhancing your integration landscape. However, this transition carries inherent risks that demand careful consideration and mitigation. In this article, we'll explore the key risks associated with a WSO2 EI to MI migration and provide insights on how to navigate them effectively.

a. Task Ownership

Ownership and involvement in each aspect of the migration must be clearly defined. For example; the DevOps team play a critical role in orchestrating the deployment and infrastructure aspects, while developers are responsible for refactoring and adapting custom components and integrations, testing teams need to ensure that migration doesn't disrupt critical workflows. Therefore, establishment of clear communication channels and collaboration between these teams to avoid misunderstandings and bottlenecks is vital to successfully complete a migration on time.

b. Changes required for the Continuous Integration/Continuous Deployment (CI/CD) Flow

The migration to Micro Integrator may necessitate significant adjustments to your CI/CD pipeline. Changes in deployment methods, artifact formats, and configuration handling may be required. Ensure that your CI/CD pipeline is updated to accommodate these changes seamlessly. Testing in a staging environment before migration is crucial to identify and address any issues in the CI/CD flow.

c. Unstructured Integration Artifacts

Unstructured integration artifacts may need to be refactored to align with the Micro Integrator's framework. This process may involve reorganizing and optimizing existing integrations for better performance and maintainability.

d. Managing External Stakeholders

Managing external stakeholders, such as partner organizations or third-party systems, to ensure comprehensive end-to-end testing can be challenging. Effective communication and coordination are vital to facilitate testing of integration flows that extend beyond your organization's boundaries.

e. Availability of integrations tests

Availability of integration tests that validate the functionality of existing integrations is crucial. Validate that your existing tests are suitable for Micro Integrator and update them as needed. Conduct rigorous testing before migration to confirm that integrations behave as expected in the new environment.

Conclusion (What is special about EADX when it comes to the migration)

After considering the points above, it is evident that selecting suitable engineering resources from the entities who are already having extensive migration experience is crucial for a successful product migration. Success in the migration process extends beyond solving technical puzzles; it also hinges on efficiently managing stakeholders and timelines. Effective stakeholder management is also pivotal for the overall success of the product migration. The core objective of this article is to provide a comprehensive understanding about the migration paradigm of WSO2 products. Even though you are not into deep technical jargons, by reading this article you will be enlightened to take most productive decisions which are beneficial for everyone around your integration echo system.

References

[1] https://wso2.com/products/support-matrix/

[2] https://apim.docs.wso2.com/en/latest/reference/config-catalog-mi/

[3] https://apim.docs.wso2.com/en/latest/observe/mi-observe/working-with-monitoring-dashboard/

[4] https://apim.docs.wso2.com/en/latest/mi-analytics/mi-elk-installation-guide/#elastic-stack-based-operational-analytics-for-micro-integrator

Talk to our experts

If you like to get in touch with migration expert at EADX for extended support, do not hesitate to contact us over below channel

sales@eadx.com