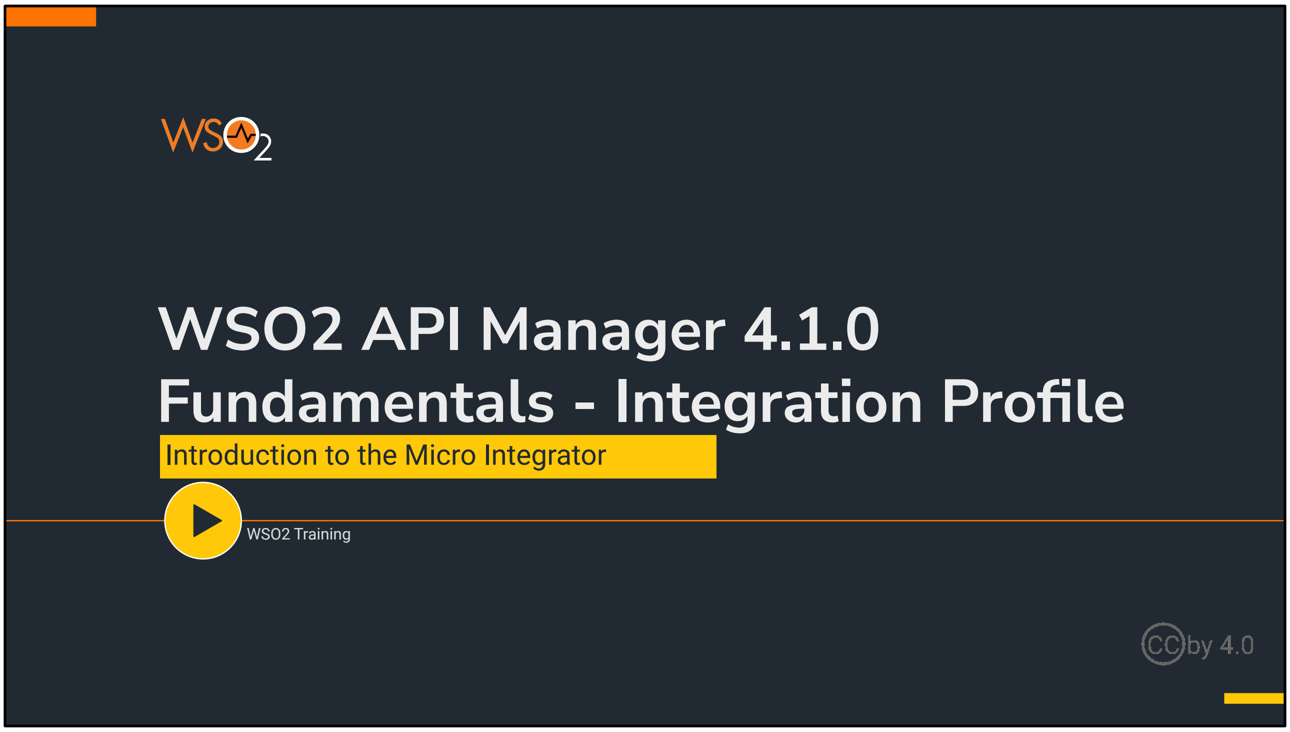


# WSO2 API Manager 4.1.0 Fundamentals - Integration Profile

Introduction to the Micro Integrator

WSO2 Training

CC by 4.0

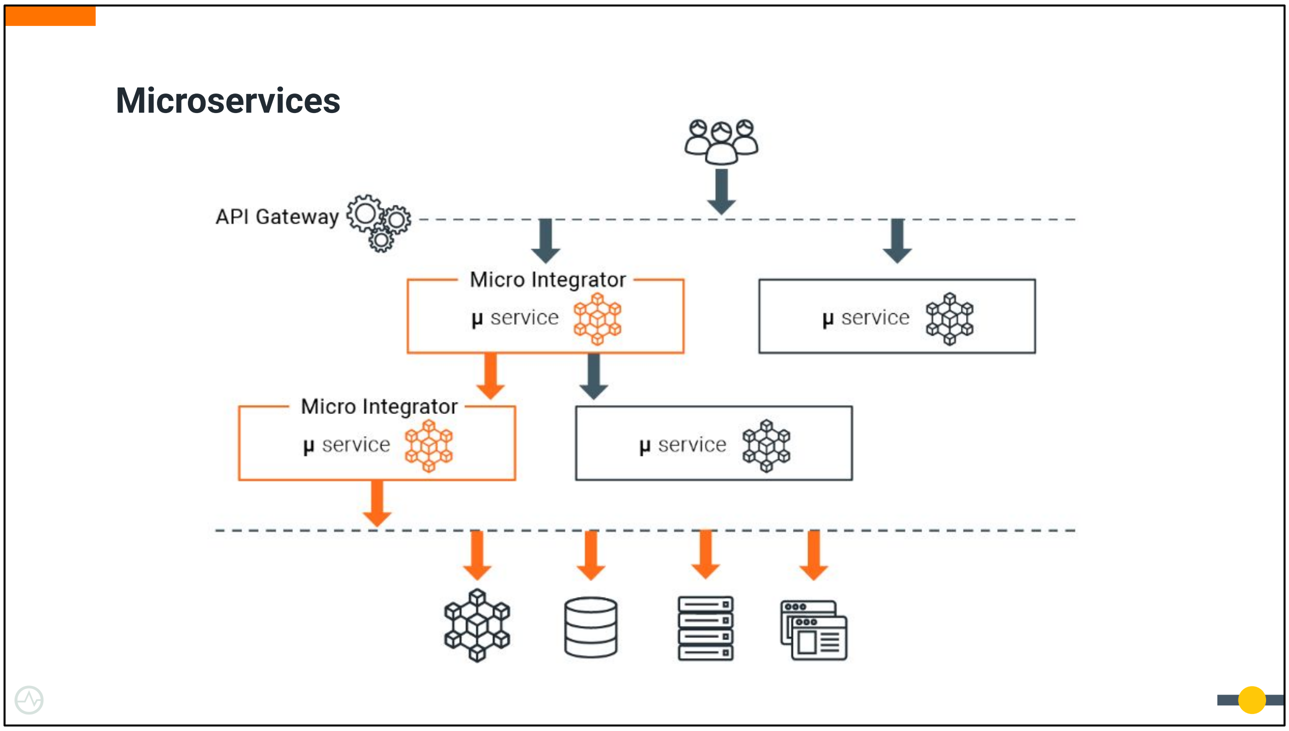
## Module Objective

At the end of this module, attendees will be able to:

- Understand the basics of WSO2 Micro Integrator.
- Understand how to implement EIP patterns with WSO2 Micro Integrator.

## WSO2 Micro Integrator

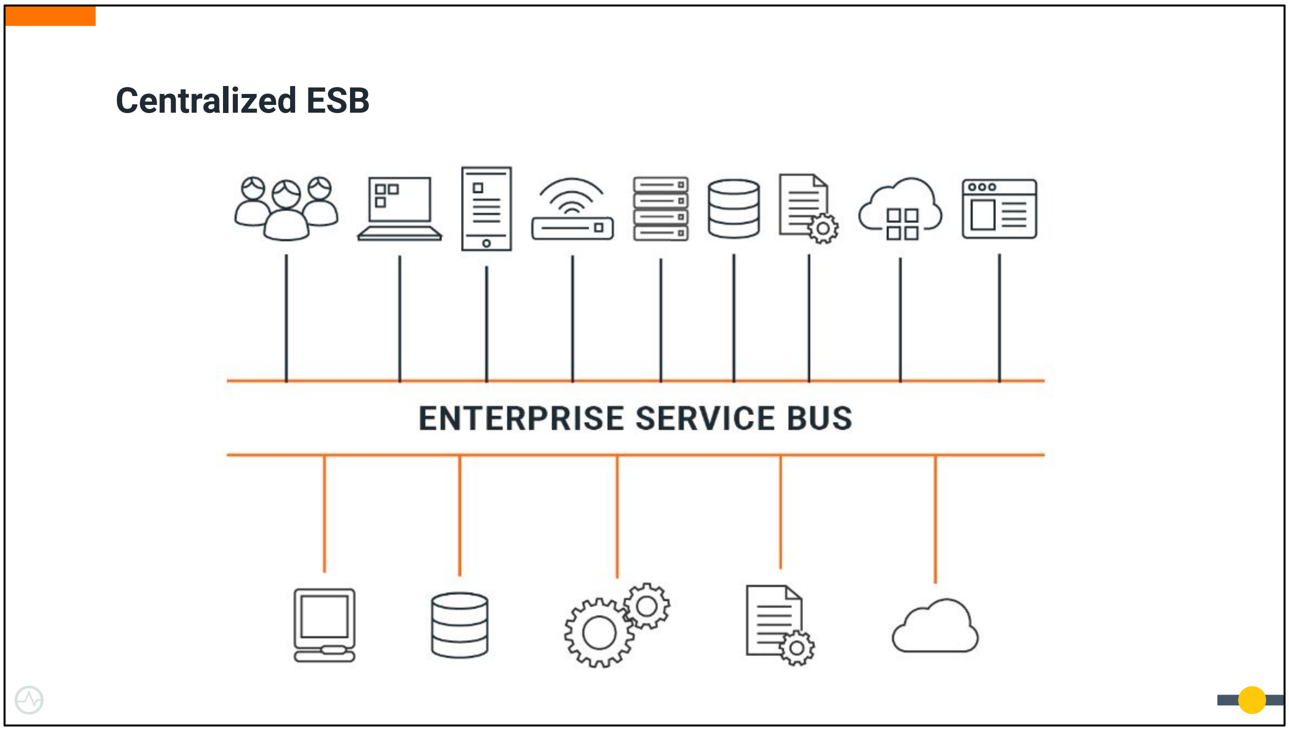
- Supports service integration, messaging, as well as data integration.
- Supports both centralized and decentralized integration architectures.



## Microservices

WSO2 Micro Integrator is light-weight and container friendly. This allows you to leverage the comprehensive enterprise messaging capabilities of WSO2 Micro Integrator in your decentralized, cloud-native integrations.

As shown above, if your organization is running on a decentralized, cloud-native integration architecture where microservices are used for integrating the various APIs, events, and systems, WSO2 Micro Integrator can easily function as your **Integration** (micro) services and **API** (micro) services.



## Centralized ESB

The heart of WSO2 Micro Integrator is an event-driven, standards-based messaging engine (the **Bus**), which supports message routing, message transformations, and other types of messaging use cases. If your organization uses an API-driven, centralized, integration architecture, WSO2 Micro Integrator can be used as the central integration layer that implements the message mediation logic connecting all the systems, data, events, APIs, etc. in your integration ecosystem.

## Low - Code Integration (WSO2 Integration Studio)

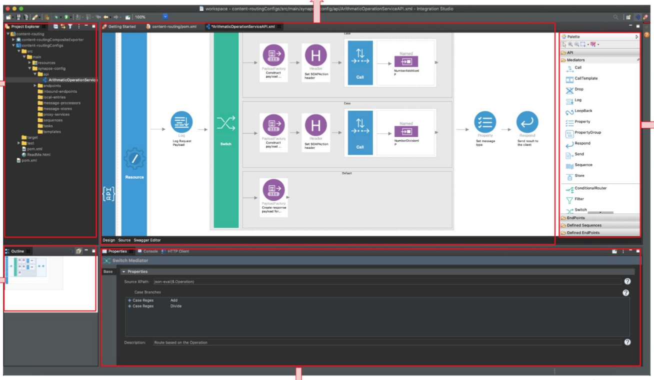
Graphical
Integration
Designing!

**Editor**

The editor allows you to drag and drop mediators, operations, and other artifacts from the Palette onto the canvas to create your integration flow. You can also switch from Design view to Source view by selecting the tabs at the bottom left of the editor.

**Project Explorer**

View of files and the folder structure within a project



**Palette**

Contains the available APIs, mediators, endpoints, connector operations, defined sequences and defined endpoints

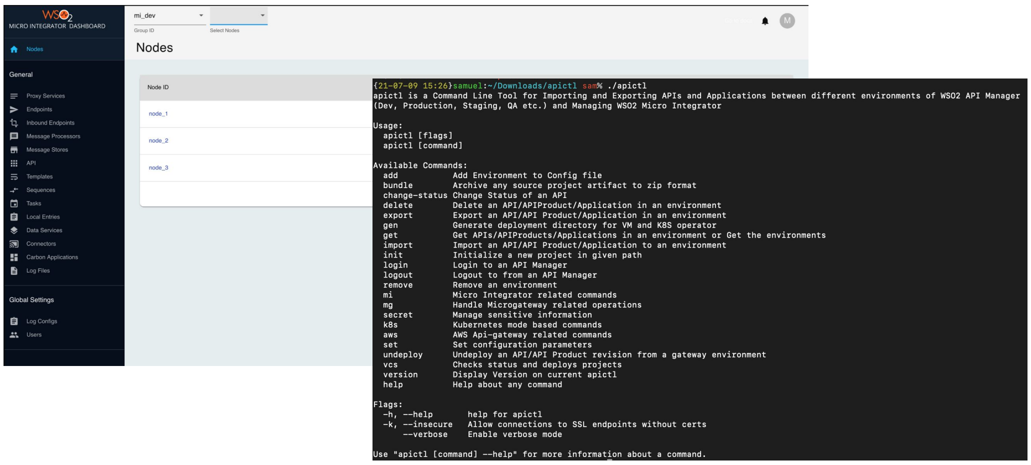
**Outline**

Outline is in sync with what you are doing in the Editor and provides an overview of the contents of the file you are working on

**Properties, Console, and HTTP Client**

List of properties of the selected item and their respective value. The Console view displays a variety of console types depending on the type of development and the current set of user settings.

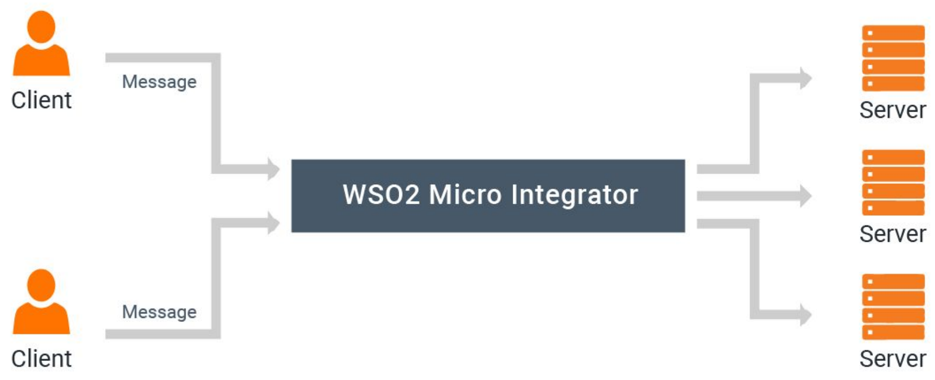
## Administration - Dashboard and CLI



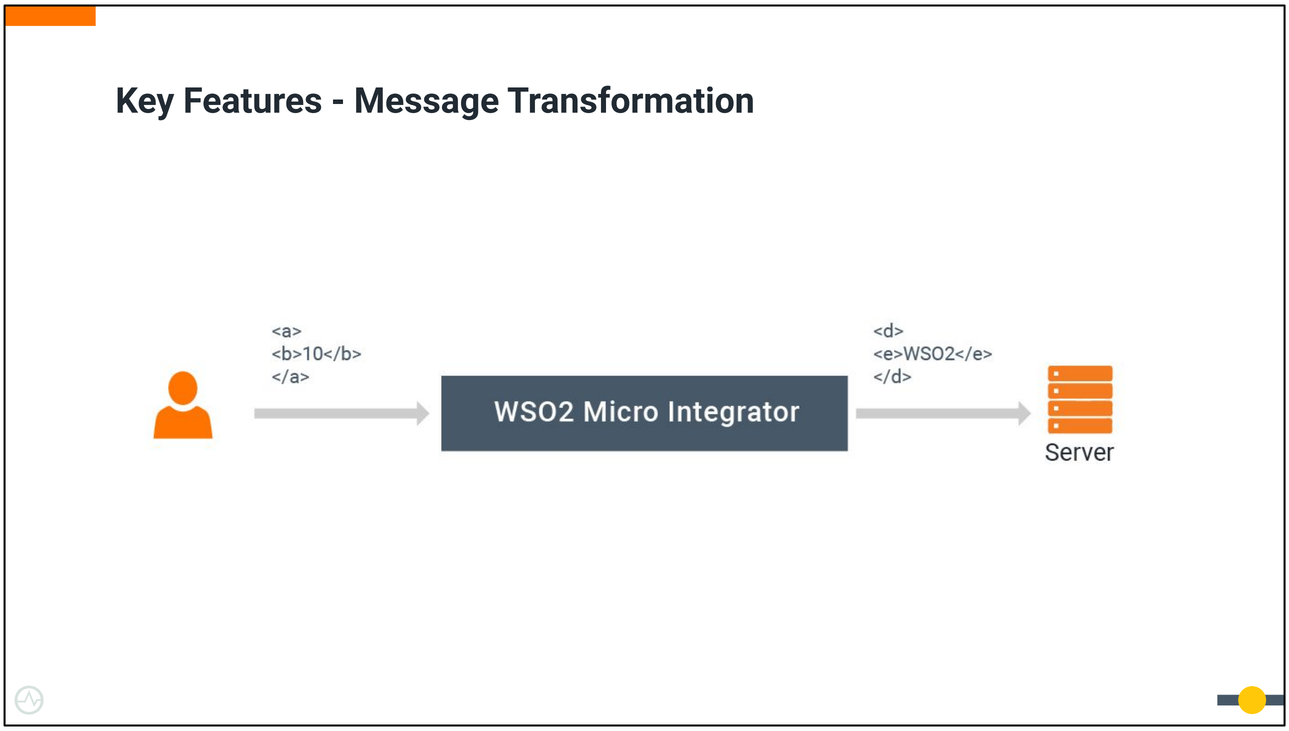
```
{21-07-09 15:26}samuel:~/Downloads/apictl sam% ./apictl
apictl is a Command Line Tool for Importing and Exporting APIs and Applications between different environments of WSO2 API Manager
(Dev, Production, Staging, QA etc.) and Managing WSO2 Micro Integrator

Usage:
  apictl [flags]
  apictl [command]

Available Commands:
  add           Add Environment to Config file
  bundle        Archive any source project artifact to zip format
  change-status Change Status of an API
  delete        Delete an API/APIProduct/Application in an environment
  export        Export an API/API Product/Application in an environment
  gen           Generate deployment directory for VM and K8S operator
  get           Get APIs/APIProducts/Applications in an environment or Get the environments
  import        Import an API/API Product/Application to an environment
  init          Initialize a new project in given path
  login         Login to an API Manager
  logout        Logout to from an API Manager
  remove        Remove an environment
  mi            Micro Integrator related commands
  mg            Handle Microgateway related operations
  secret        Manage sensitive information
  k8s           Kubernetes mode based commands
  aws           AWS Api-gateway related commands
  set           Set configuration parameters
  undeploy      Undeploy an API/API Product revision from a gateway environment
  vcs           Checks status and deploys projects
  version       Display Version on current apictl
  help          Help about any command

Flags:
  -h, --help       help for apictl
  -k, --insecure   Allow connections to SSL endpoints without certs
      --verbose    Enable verbose mode

Use "apictl [command] --help" for more information about a command.
```

You can perform administration tasks in the Micro Integrator by using the CLI and the Dashboard. You can also perform operations by directly calling the management API.

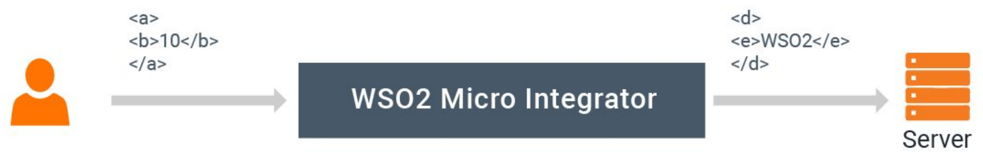
## Key Features - Message Routing



When the Micro Integrator processes individual messages from clients, it can route messages based on the content of the message.
This is known as content-based routing.



## Key Features - Message Transformation



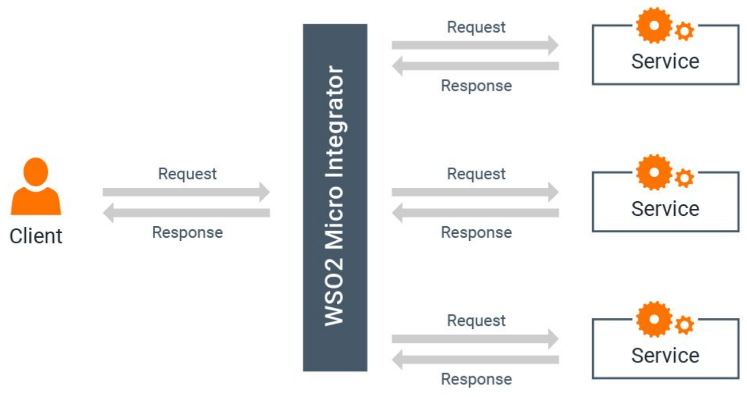
```
<a>
<b>10</b>
</a>
```

```
<d>
<e>WSO2</e>
</d>
```

Another key and powerful feature of the Micro Integrator is message transformation.

This includes manipulating messages in any required manner, adding/removing content from messages, converting a message to a completely different message format, and even validating messages based on the available validation mechanisms of the message format.

## Key Features - Service Orchestration



Service Orchestration is the process of exposing multiple fine-grained services using a single coarse-grained service. The service client will only have access to a single coarse-grained service, which encapsulates the multiple fine-grained services that are invoked in the process flow.

There are two distinct types of service orchestration:

- Synchronous service orchestration
- Asynchronous service orchestration

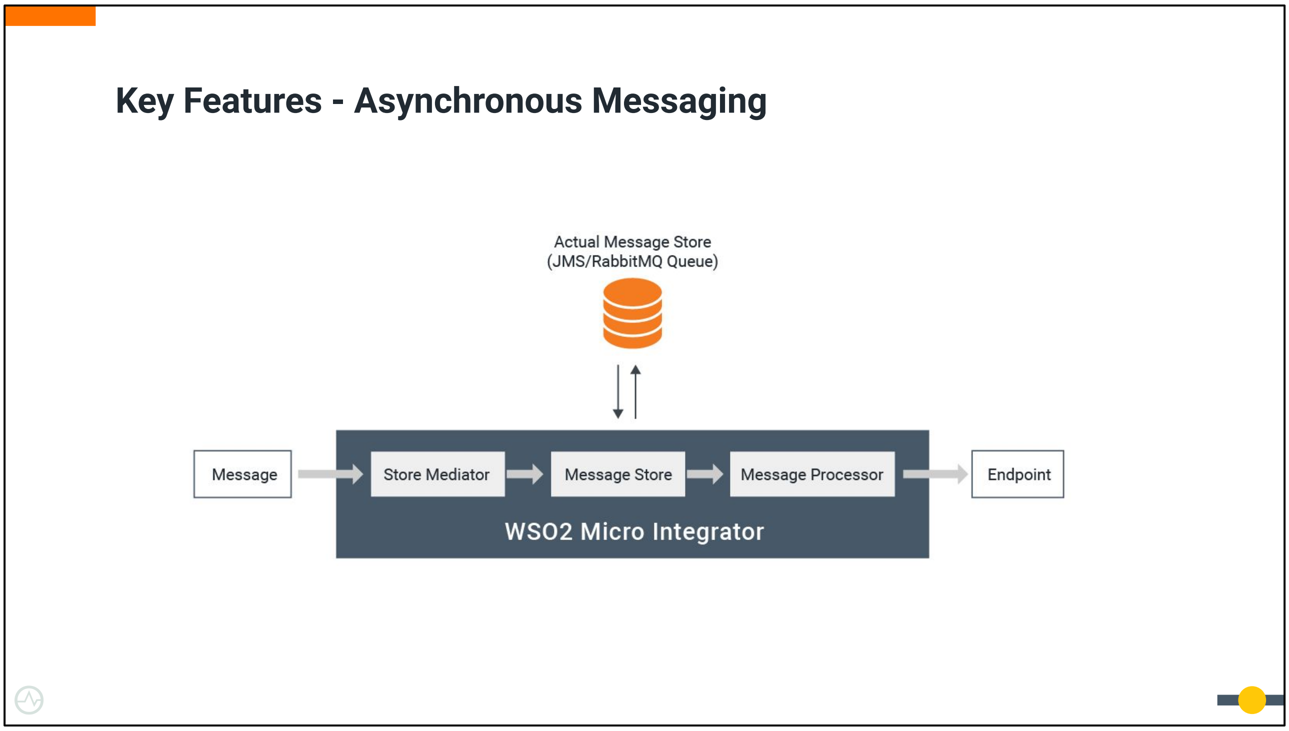
In both the above orchestration approaches, the WSO2 Micro Integrator can interact with services using two different patterns (depending on the use case):

**Service chaining**

Multiple services that need to be orchestrated are invoked one after the other in a synchronous manner. The input to one service is dependant on the output of another service. Invocation of services and input-output mapping is handled by the service orchestration layer (which is the WSO2 Micro Integrator).

**Parallel service invocations**

Multiple services are invoked simultaneously without any blocking until a response is received from another service.



## Key Features - Asynchronous Messaging

Asynchronous messaging is a communication method wherein the system puts a message in a message queue and does not require an immediate response to continue processing. Asynchronous messaging is useful for the following:

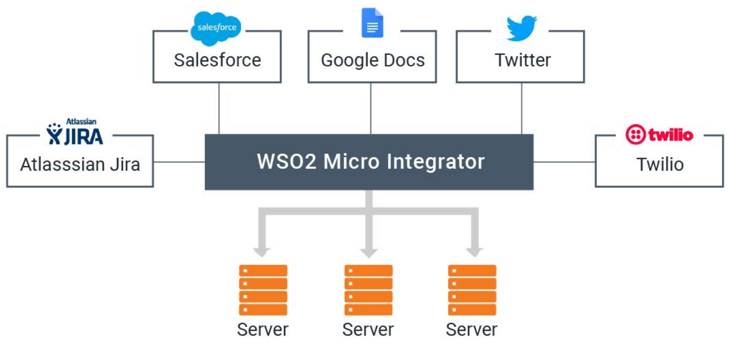
- Delegate the request to some external system for processing
- Ensure delivery of a message to an external system
- Throttle message rates between two systems
- Batch processing of messages

Note the following about asynchronous message processing:

- Asynchronous messaging solves the problem of intermittent connectivity. The message receiving party does not need to be online to receive the message as the message is stored in a middle layer. This allows the receiver to retrieve the message when it comes online.
- Message consumers do not need to know about the message publishers. They can operate independently.

Disadvantages of asynchronous messaging includes the additional component of a message broker or transfer agent to ensure the message is received. This may affect both performance and reliability. There are various levels of message delivery reliability grantees from publisher to broker and from broker to subscriber. Wire level protocols like AMQP and MQTT can provide those.

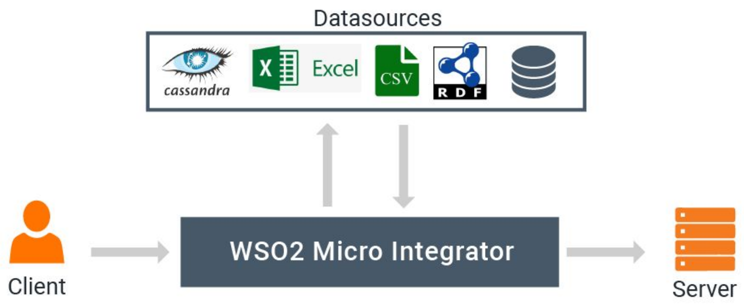
## Key Features - Connecting Web APIs/Cloud Services



One of the most expected features from an integration solution is 'Hybrid-Cloud Integration', i.e., the ability to connect with third-party applications via public APIs that are exposed by various application developers. These external applications could either be in the cloud (SaaS) or on-premise. WSO2 Micro Integrator enables this capability by means of its wide range of connectors.

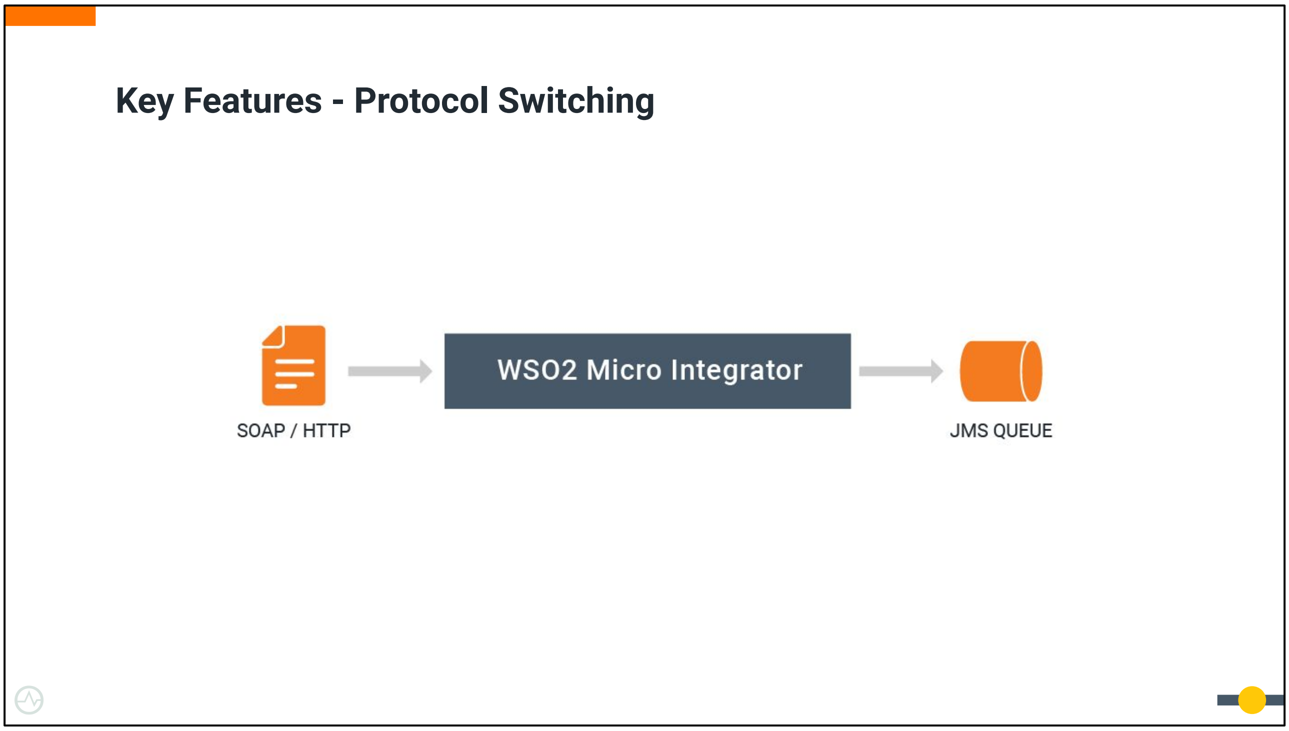
WSO2 Micro Integrator supports more than 150 connectors, where Salesforce, Gmail, Amazon S3, are among the most popular. Connectors can be downloaded from the WSO2 connector store. It is also possible to write your own custom connector to integrate WSO2 Micro Integrator with a new system.

## Key Features - Data Integration

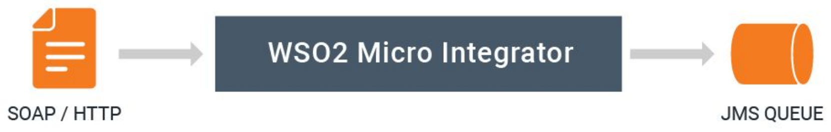


Data integration is an important part of an integration process. For example, consider a typical integration process that is managed using the Micro Integrator: Data stored in various, disparate datasources are required in order to complete the integration use case.

The data services functionality that is embedded in the Micro Integrator can decouple the data from the datasource layer and exposing them as data services. The main integration flow defined in the Micro Integrator will then have the capability of managing the data through the data service. Once the data service is defined, you can manipulate the data stored in the datasources by invoking the relevant operation defined in the data service. For example, you can perform the basic CRUD operations as well as other advanced operations.

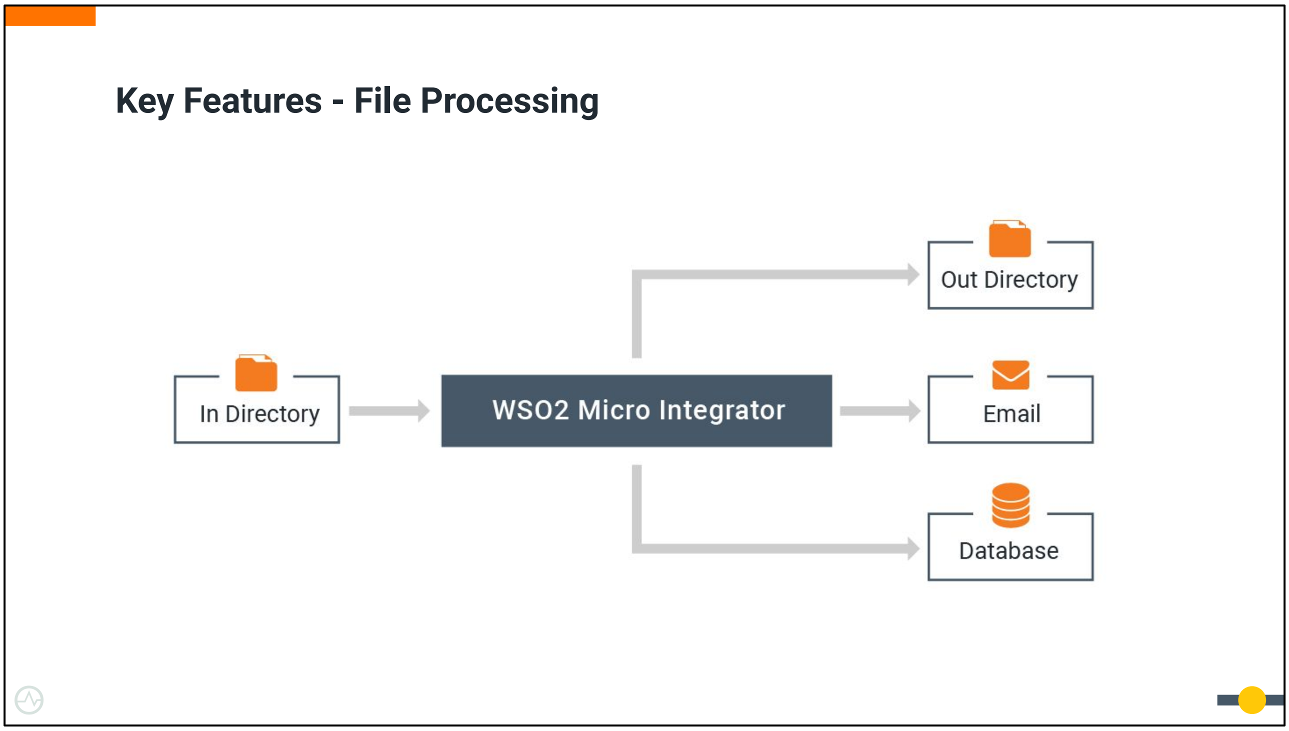


## Key Features - Protocol Switching

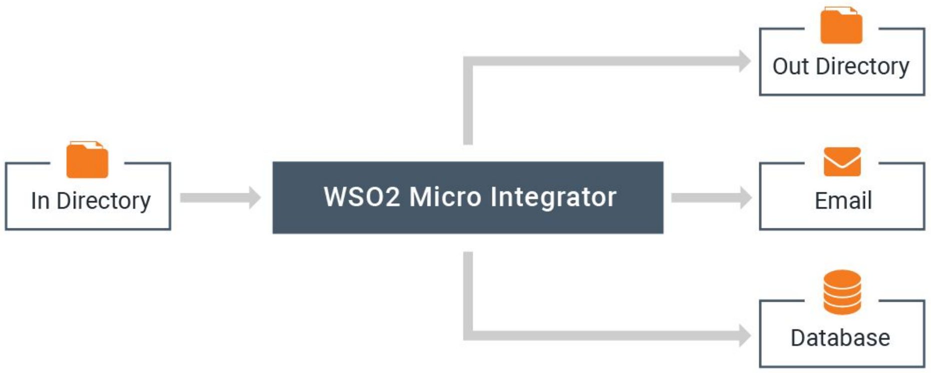


Another useful feature of the Micro Integrator is the ability to switch protocols during messaging.

This makes it possible to take a message that comes in via HTTP and send it out to a JMS queue.
Another example that demonstrates the power of protocol switching + transformation is the ability to take messages that come in from one protocol like HTTP, take the business content of the message, and then send the content out in a completely different message format and protocol.

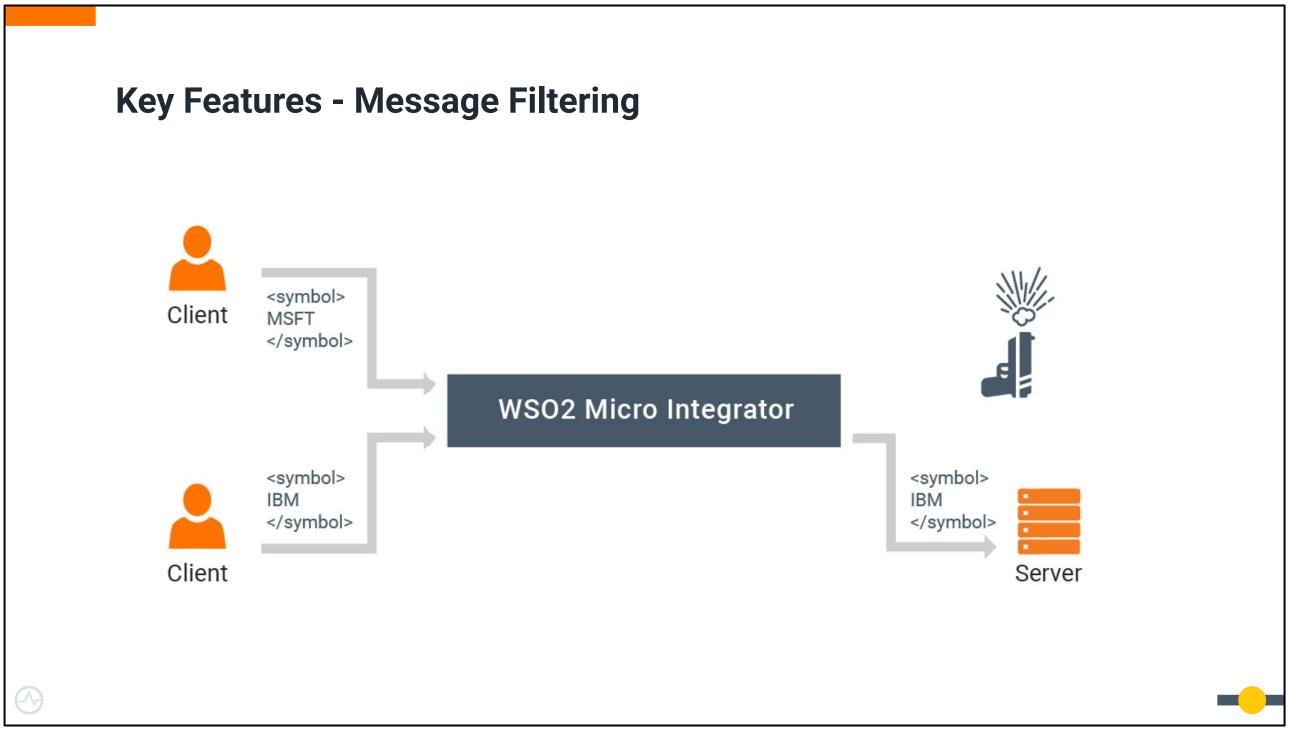


## Key Features - File Processing

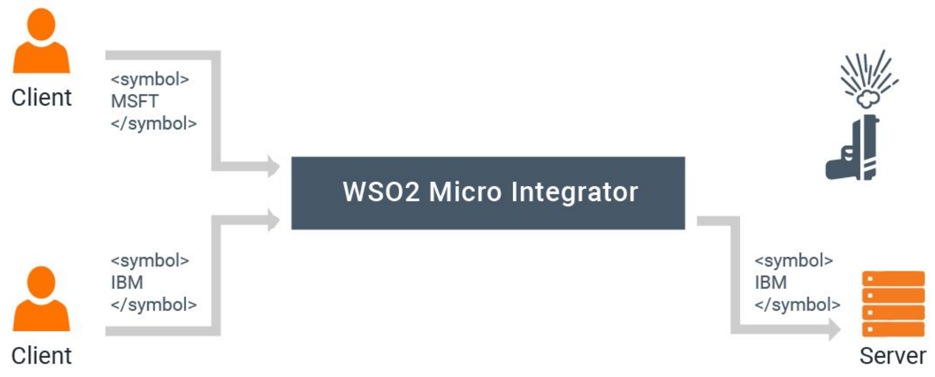


In many business domains, there are different use cases related to managing files. Also, there are file-based legacy systems that are tightly coupled with other systems. These files contain huge amounts of data, which requires a big effort for manual processing. It is not scalable with an increase in system load. This leads us to the requirement of automating the processing of files. The WSO2 Micro Integrator enables the following file processing capabilities:

- Reading, Writing, and Updating files:
  Files can be located in the local file system or a remote location that can be accessed over protocols such as FTP, FTPS, SFTP, SMB. Therefore, the system used to process those files should be capable of communicating over those protocols.
- Process data
  The system should be capable of extracting relevant information from the file. For example, if it is required to process XML files, the system should be capable of executing an XPath on the file content and extract relevant information.
- Execute some business logic
  The system should be capable of performing actions that are required to construct a business use case. It should be capable of taking decisions and sending processed information to other systems over different communication protocols.



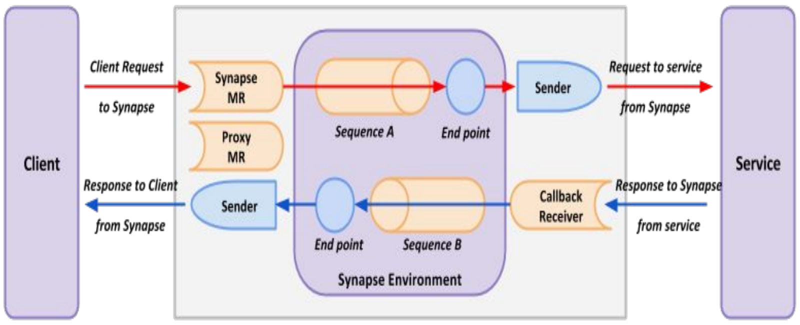
## Key Features - Message Filtering



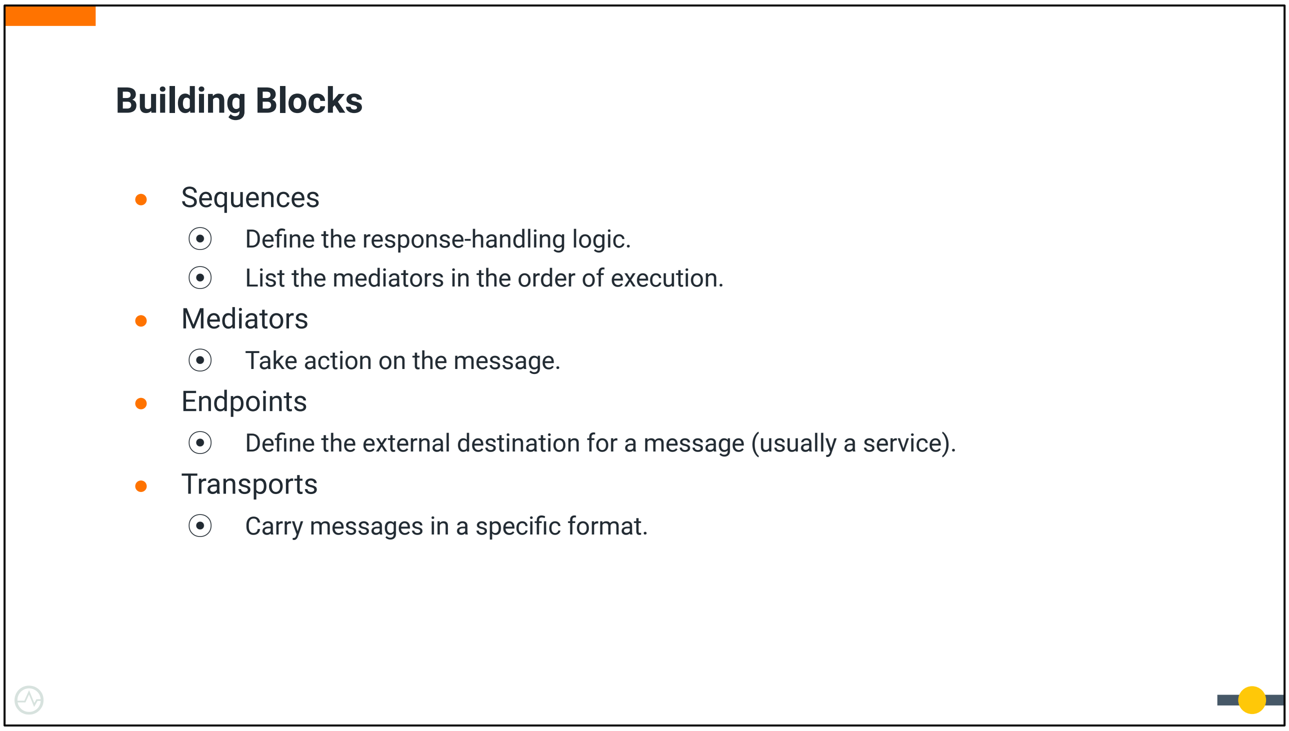
Another important feature is message filtering.
Filtering is where the Micro Integrator filters messages based on the content of the message.

Filtering is not just about filtering messages that the Micro Integrator decides to send out. It also allow more complex logic where messages can be filtered and sent in different mediation flows.

## Building Blocks



Synapse MR = Synapse Message Receiver
Proxy MR = Proxy Message Receiver



## Building Blocks

- Sequences
  - Define the response-handling logic.
  - List the mediators in the order of execution.
- Mediators
  - Take action on the message.
- Endpoints
  - Define the external destination for a message (usually a service).
- Transports
  - Carry messages in a specific format.

Sequences are the configuration component for mediators. Sequences allow you to organize the mediators for implementing pipes and filter patterns.

Mediators are individual processing units that perform a specific function such as sending or filtering messages. The Micro Integrator includes a comprehensive mediator library that provides functionality for implementing widely used enterprise integration patterns (EIPs). You can also easily write a custom mediator to provide additional functionality using various technologies such as Java, scripting, and Spring.

An endpoint defines an external destination (such as a service) for a message. An endpoint can be specified as an address endpoint, WSDL endpoint, a load balancing endpoint, etc.
An endpoint is defined independently of transports, allowing you to use the same endpoint with multiple transports. When you configure a message mediation sequence or a proxy service to handle the incoming message, you specify the transport to use and the endpoint where the message will be sent.

Transports - Carry messages in a specific format.

The Micro Integrator supports all the widely used transports including HTTP/s, JMS, and VFS, and domain-specific transports like FIX. All WSO2 transports are directly or indirectly based on the Apache Axis2 transports framework.

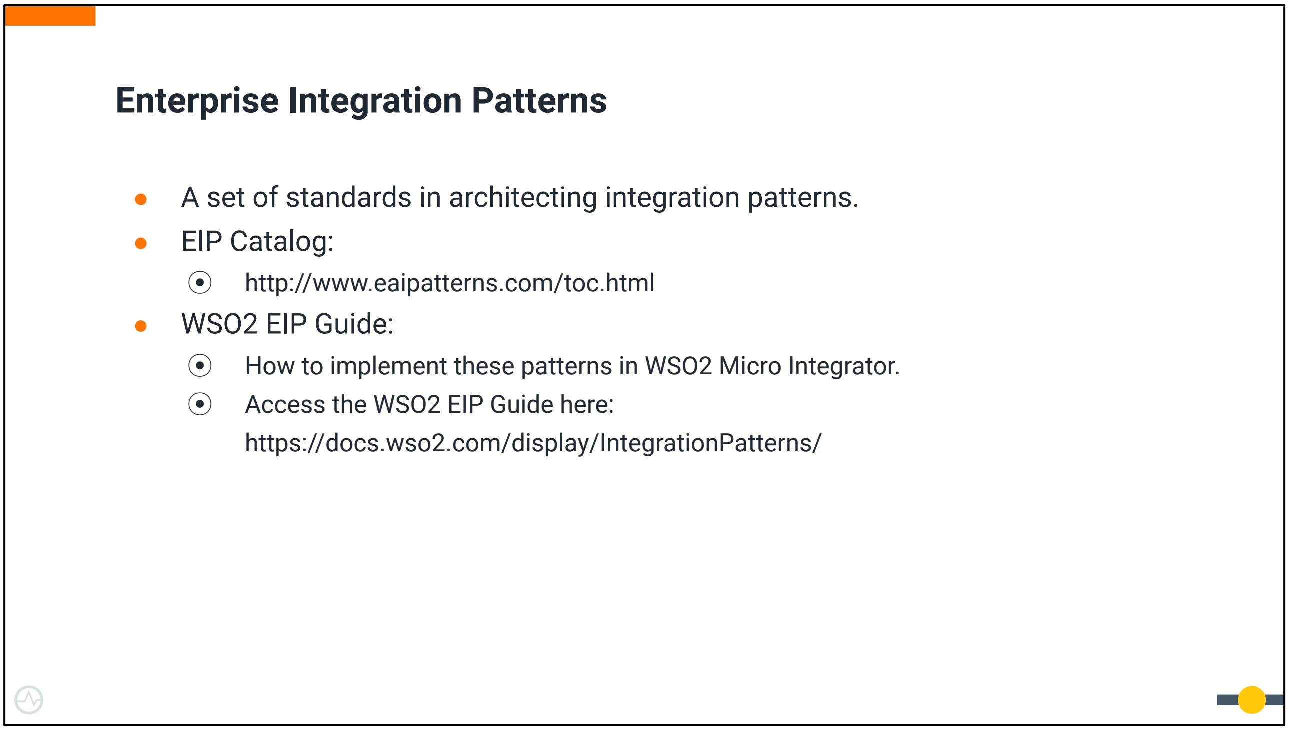
## Let’s try it out!

Getting Started with WSO2 Micro Integrator

Getting Started with WSO2 Integration Studio

CC BY 4.0

## Enterprise Integration Patterns (EIPs)



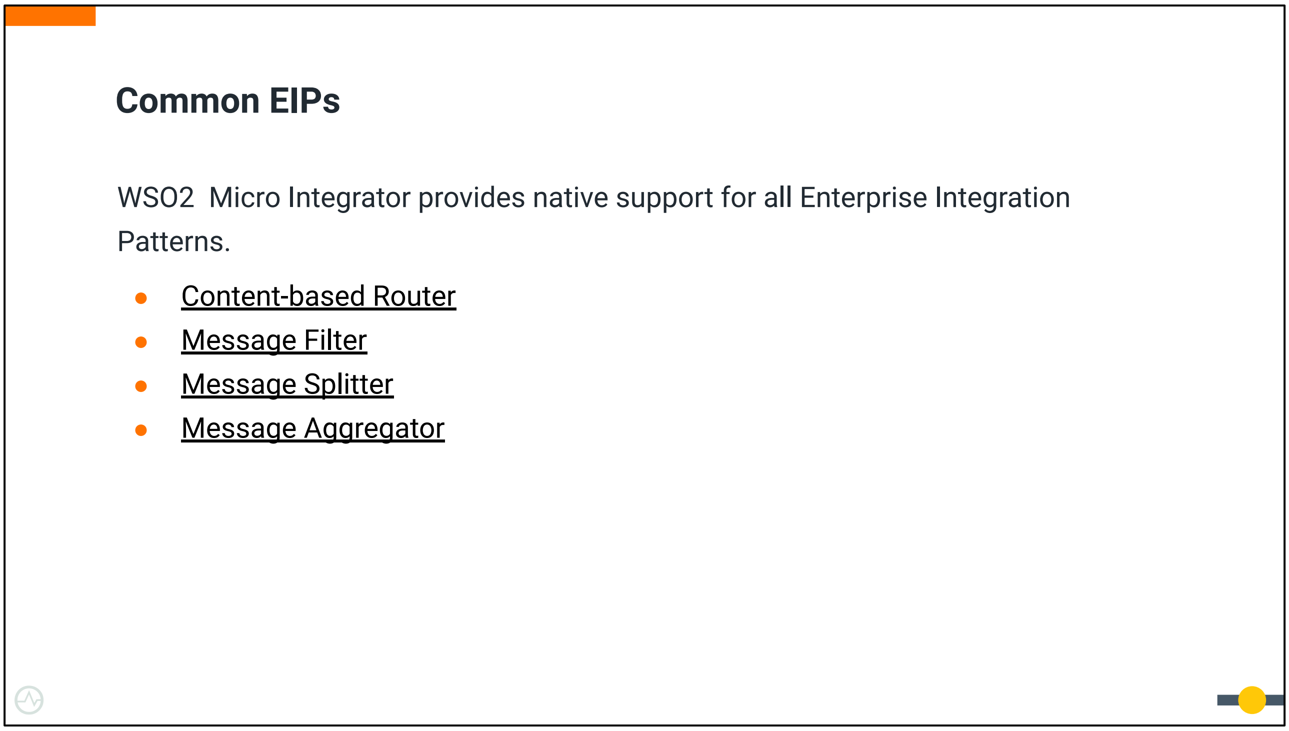
## Enterprise Integration Patterns

- A set of standards in architecting integration patterns.
- EIP Catalog:
  - http://www.eaipatterns.com/toc.html
- WSO2 EIP Guide:
  - How to implement these patterns in WSO2 Micro Integrator.
  - Access the WSO2 EIP Guide here: https://docs.wso2.com/display/IntegrationPatterns/

Go to http://www.eaipatterns.com/toc.html

Take a look at the descriptions of the 65 patterns.
EIP Book has more details on each.

Go to
https://docs.wso2.com/display/EIP/Enterprise+Integration+Patterns+with+WSO2+Enterprise+Integrator and walk through the EIP patterns at a high level. Then drill down and see how to use each.



## Common EIPs

WSO2  Micro Integrator provides native support for all Enterprise Integration Patterns.

- Content-based Router
- Message Filter
- Message Splitter
- Message Aggregator

Review these common EIPs, and understand how the Micro Integrator can handle each of them.