**Mule ESB**

Mule ESB is lightweight java Enterprise Service Bus. It is a platform that help developer connect application together quickly and easily without requiring custom code.

Mule ESB help to easily “**create services”** or host existing ones, say for example using existing POJO over and over again, it also help in **“service mediation”, i.e** when you change something the user dosen’t need to know, it also does **“message routing”, “data transformation”**.

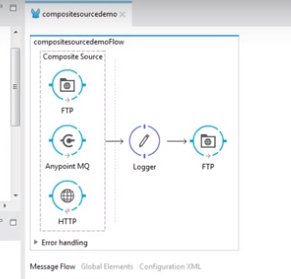
**#[payload] –** suppose flow is like

FTP-Logger(#[payload])-XML To JSON-Logger(#[payload])

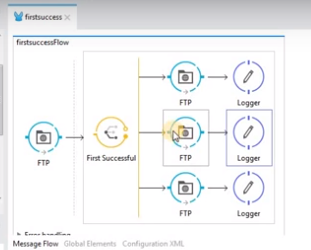
First Logger will print whatever comes from FTP, Second logger print the JSON which we got as output

**Composite Source**

A condition can arrive when we need input from different sources going to a single path. For Example



**FIRST SUCCESSFUL**



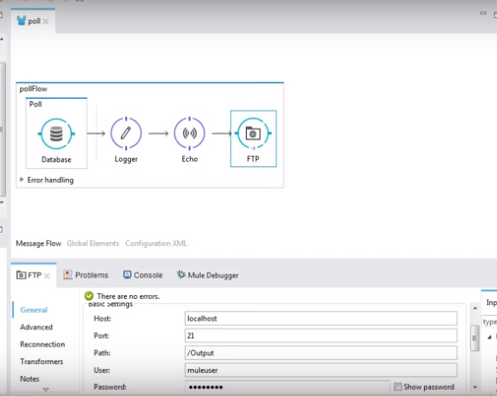
It tries to put message one by one , whichever get success is executed.

Say it successfully place it in 2nd FTP, so it won’t go to 3rd FTP.

**XSLT**

Extensible Stylesheet Language.

**POLL AND ECHO**



**Poll** - While some connectors, such as [HTTP](https://docs.mulesoft.com/mule-user-guide/v/3.7/http-connector) and [FTP](https://docs.mulesoft.com/mule-user-guide/v/3.7/ftp-connector), utilize a polling process to actively retrieve messages from an external resource, most message processors in Mule are triggered when called by a previous element in a flow. If you want to arrange for a message processor to actively call a resource at regular intervals, use a Poll scope.

**ECHO** - **Purpose:** Use Echo to echo or display the message payload. Note that the component transforms the payload into a String.

**BATCH AND BATCH-EXECUTE**

Batch Processing

***Enterprise, CloudHub***

Mule possesses the ability to process messages in batches. Within an application, you can initiate a batch job which is a block of code that splits messages into individual records, performs actions upon each record, then reports on the results and potentially pushes the processed output to other systems or queues. This functionality is particularly useful when working with streaming input or when engineering "near real-time" data integration between SaaS applications.

**EXPRESSION LANGUAGE VS SCRIPTING LANGUAGE**

