



# Edge Microgateway

## Custom Plugins

# Agenda

- Custom Plugin Overview
- Custom Plugin Lab

# Plugin Overview

- A plugin is a Node.js module that can implement one or more of following Microgateway events.
- Plugins also allow you to extend the Microgateway and add new features.

Request events to override:

- `onrequest`, `ondata_request`, `onend_request`, `onclose_request`, `onerror_request`

Response events to override:

- `onresponse`, `ondata_response`, `onend_response`, `onclose_response`, `onerror_response`

# Plugin Event Architecture - Request

onrequest

Called at start of request

Access headers, URL, query params, HTTP method

Call next()

ondata\_request

Called when a chunk of data is received

Call next(null, data)

onend\_request

Called when all the data has been received

Call next(null, data)

onclose\_request

Called when the client has closed the connection

Call next()

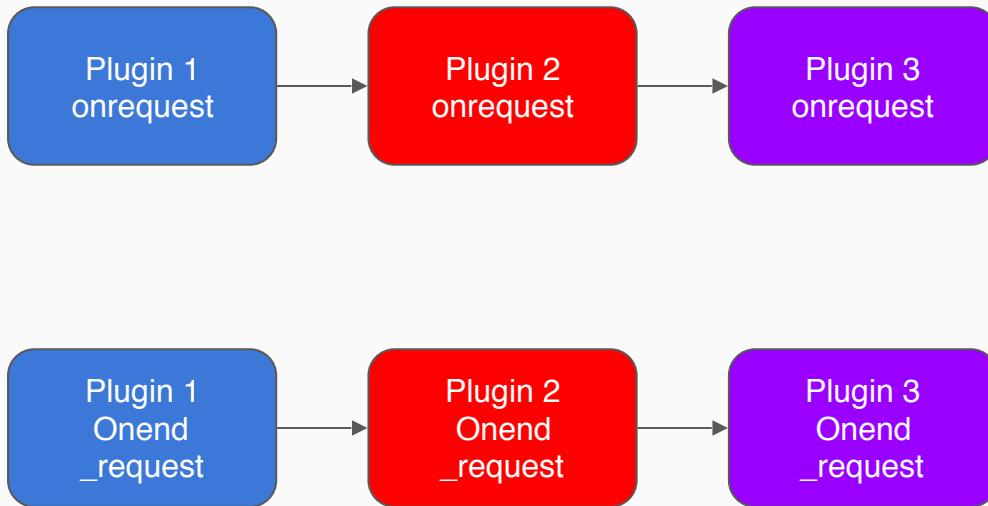
onerror\_request

Called when there is an error receiving the client request

Call next()

# Plugin Event Model - Request Flow

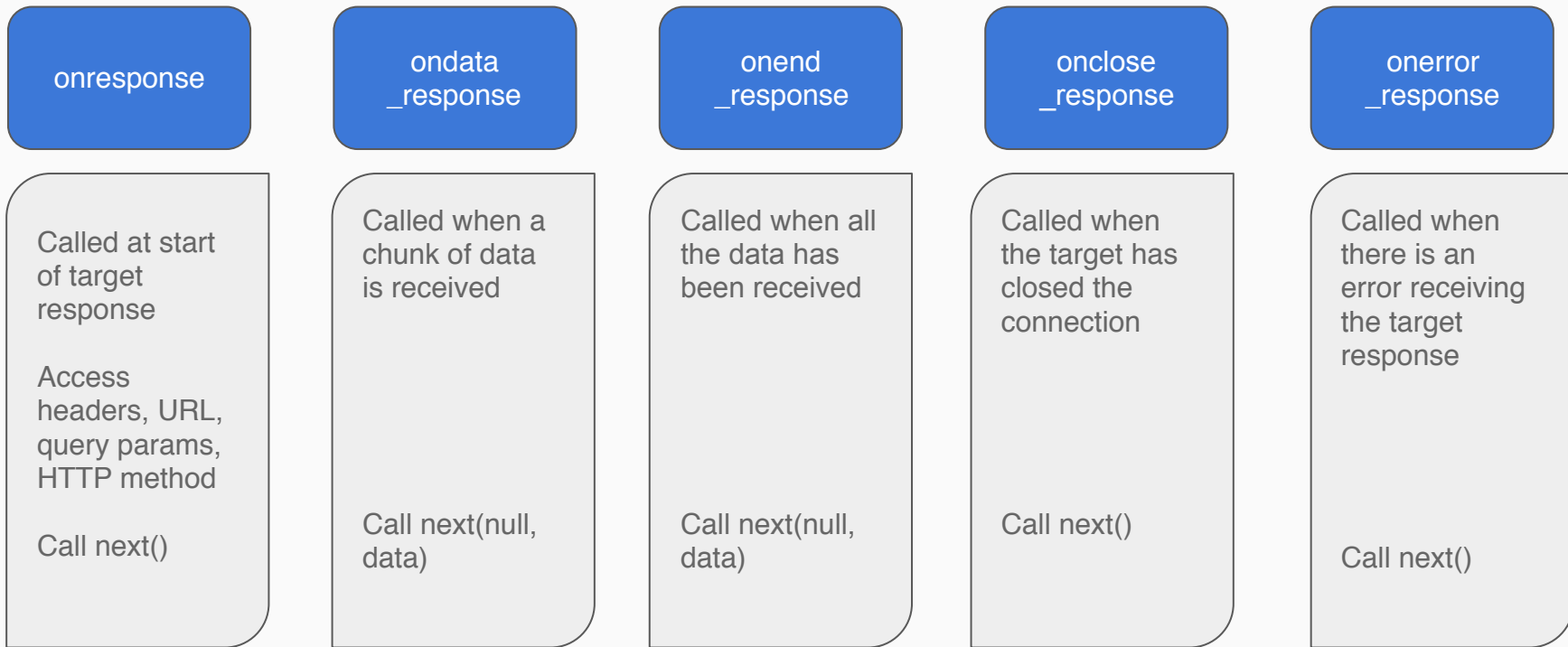
```
edgemicro:  
...  
  plugins:  
    sequence:  
      - plugin1  
      - plugin2  
      - plugin3
```



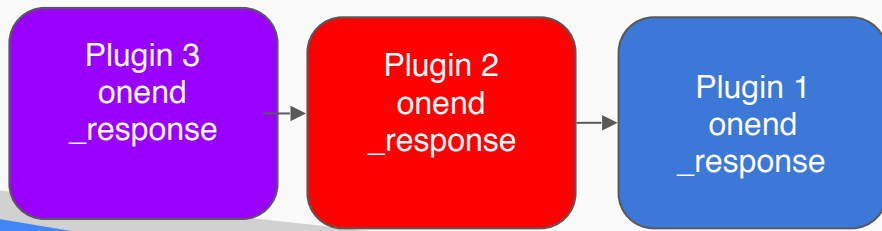
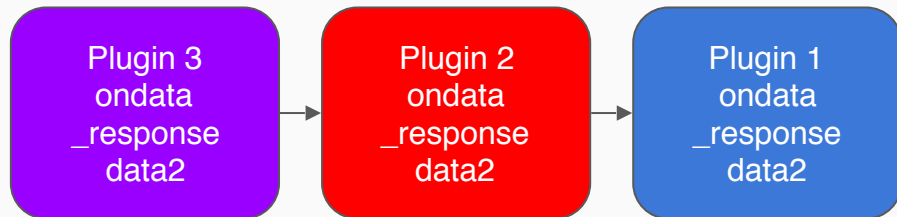
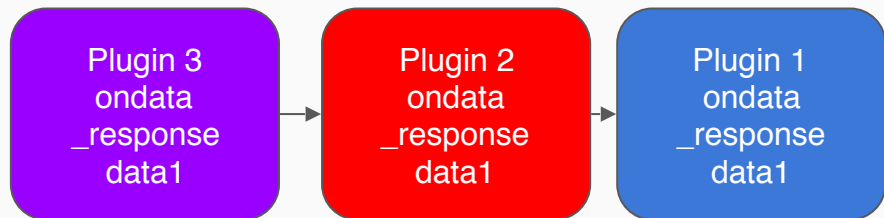
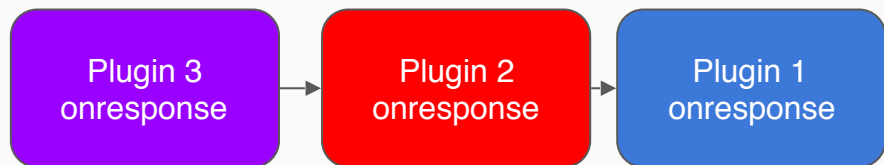
# Plugin Event Model - Request Flow

```
^[[A gateway:main selected proxy http://localhost:8087 with base path /edgemicro_hello for request
path /edgemicro_hello/hello +2ns
gateway:main sourceRequest +3ns ce4005b0-6e2f-11e7-9726-9991b1fd7e348 GET /edgemicro_hello/hello
plugin:plugin1 plugin-1: onRequest +2ns
plugin:plugin2 plugin-2: onRequest +1ms
gateway:main targetRequest +6ns ce4005b0-6e2f-11e7-9726-9991b1fd7e348 GET localhost:8087 /hello
gateway:main plugin analytics does not provide handler function for end_request +4ns
plugin:plugin1 plugin-1: onend_request +11ms
plugin:plugin2 plugin-2: onend_request +0ms
gateway:main targetResponse +5ms ce4005b0-6e2f-11e7-9726-9991b1fd7e348 200
plugin:plugin2 plugin-2: onResponse null +2ms
plugin:plugin1 plugin-1: onResponse null +1ms
gateway:main plugin analytics does not provide handler function for response +1ms
gateway:main [ null, null, null ] +0ms
gateway:main req data +3ms 22
plugin:plugin2 plugin-2: ondata_response {
plugin:plugin2   "hello": "world"
plugin:plugin2 } +3ms
plugin:plugin1 plugin-1: ondata_response {
plugin:plugin1   "hello": "world"
plugin:plugin1 } +0ms
gateway:main plugin analytics does not provide handler function for data_response +0ms
plugin:plugin2 plugin-2: onend_response +2ms
plugin:plugin1 plugin-1: onend_response +0ms
gateway:main plugin analytics does not provide handler function for end_response +2ms
analytics flushing 1 records. 0 records remaining. +98ms
analytics returned 1 records to buffer +73ms
Checking for change in configuration
```

# Plugin Event Architecture - Response



# Plugin Event Model - Response Flow



```
edgemicro:  
...  
  plugins:  
    sequence:  
      - plugin1  
      - plugin2  
      - plugin3
```



# Plugin Event Model - Request Flow

```
[[[A gateway:main selected proxy http://localhost:8087 with base path /edgemicro_hello for request
path /edgemicro_hello/hello +2ns
gateway:main sourceRequest +3ns ce4085b0-6e2f-11e7-9726-9991b1fd7e348 GET /edgemicro_hello/hello
plugin:plugin1 plugin-1: onRequest +2ns
plugin:plugin2 plugin-2: onRequest +1ms
gateway:main targetRequest +6ns ce4085b0-6e2f-11e7-9726-9991b1fd7e348 GET localhost:8087 /hello
gateway:main plugin analytics does not provide handler function for end_request +4ns
plugin:plugin1 plugin-1: onend_request +11ms
plugin:plugin2 plugin-2: onend_request +0ms
gateway:main targetResponse +5ms ce4085b0-6e2f-11e7-9726-9991b1fd7e348 200
plugin:plugin2 plugin-2: onResponse null +2ms
plugin:plugin1 plugin-1: onResponse null +1ms
gateway:main plugin analytics does not provide handler function for response +1ms
gateway:main [ null, null, null ] +0ms
gateway:main req data +3ms 22
plugin:plugin2 plugin-2: ondata_response {
plugin:plugin2   "hello": "world"
plugin:plugin2 } +3ms
plugin:plugin1 plugin-1: ondata_response {
plugin:plugin1   "hello": "world"
plugin:plugin1 } +0ms
gateway:main plugin analytics does not provide handler function for data_response +0ms
plugin:plugin2 plugin-2: onend_response +2ms
plugin:plugin1 plugin-1: onend_response +0ms
gateway:main plugin analytics does not provide handler function for end_response +2ms
analytics flushing 1 records. 0 records remaining. +98ms
analytics returned 1 records to buffer +73ms
Checking for change in configuration
```

# Plugin Setup

- Custom plugins are stored in the folder below

**[prefix]/npm/lib/node\_modules/edgemicro/plugins**

- A new folder should be created in the 'plugins' directory
- Initialize the plugin directory with the following command

```
cd [prefix]/npm/lib/node_modules/edgemicro/plugins
mkdir myplugin
cd myplugin
npm init
```

- Custom plugin folder should contain a file named [index.js](#), which implements one or more of the Microgateway request or response events.

# Update Edge Microgateway Configuration

Edgemicro gateway [configuration](#) (partial view) is stored in the following file:

```
~/.edgemicro/org-env-config.yaml
```

Custom plugins must be included in the sequence section.

```
edgemicro:
  ...
  plugins:
    sequence:
      - oauth
      - myplugin
```

# Plugin Authoring



Things to remember when writing a plugin:

Request / response handlers must call **next()** when done processing. If not, processing will stop and the request will hang.

The first argument to **next()** may be an error which will cause processing to terminate.

The **ondata\_** and **onend\_** handlers must call **next()** with a second argument containing the data to be passed to the target or the client. It can be null if the plugin is buffering and does not have enough data to transform at the moment.

A single instance of the plugin is used to service all requests and responses. If per-request state needs to be retained between handler calls, it can be saved in a property added to the supplied **request** object, whose lifetime is the duration of the API call.

Catch all errors and call **next()** with the error. If not, the API call will hang.

Do not perform compute-intensive tasks in the main thread as this can adversely affect performance.

# Microgateway Labs

Complete the following Edge Microgateway Lab.

- [Lab](#) - Create a Custom Plugin

THANK YOU