

Implementing a Plug-In Service for OceanTEA

Custom functionality can be included into the user interface of OceanTEA by implementing a plug-in (micro-)service. After registering such a service with the OceanTEA API gateway, its functionality can be accessed via a new tab in OceanTEA.

An example of such a plug-in service is the OceanTEA Coral Analysis service,¹ which offers an interactive illustration of research results to accompany a scientific paper.

1 Implement Your Service

You can use any technology to develop a plug-in service for OceanTEA as long as it provides a RESTful API accessible via HTTP which adheres to the following rules:

1. All REST endpoints your service provides must be prefixed with a fixed `APIPrefix` such as `/coral_analysis`:
`<APIPrefix>/path/to/your/endpoint`
In this way, the API gateway of OceanTEA knows which requests are supposed to be routed to your service.
2. Your service must provide an HTML fragment at the URL
`<APIPrefix>/static/index.html`
which renders your tab within OceanTEA. This fragment is included into the body of the main view template of OceanTEA.²
3. Any static content that you need to include into your view must be served by your service at the URL:
`<APIPrefix>/static/<pathToContent>`
From within your `index.html` fragment, any static content is to be addressed using exactly the same *relative* URL `<APIPrefix>/static/<pathToContent>` (this results in the requests being routed through the API gateway).

¹<https://github.com/a-johanson/oceantea-coral-analysis>

²<https://github.com/a-johanson/oceantea-visualization-gateway/blob/master/views/layouts/main.handlebars>

2 Integrate Your Service With OceanTEA

To make your service known to OceanTEA, you have to register it with the OceanTEA API gateway. This is done by adding an entry to the `plugins` array in `plugins.js`.³ Such an entry needs to be structured as follows:⁴

```
{
  tabName: "P. arborea Activity",
  guiURL: "/p_arborea_activity.html",
  apiPrefix: "/coral_analysis",
  serviceName: "oceantea-coral-analysis",
  serviceHost: "localhost",
  servicePort: "3340",
  templateParameters: {
    libD3js : true,
    libCanvasPlot: true
  }
}
```

The field `tabName` is the text to be displayed on your plug-in tab in the OceanTEA GUI. `guiURL` is the URL at which your tab is located, and `apiPrefix` is the API prefix discussed in the previous section. `serviceName` is a unique identifier of your service to be used, for example, with service registries and with Docker. `serviceHost` and `servicePort` describe how to connect to your service.

`templateParameters` are additional parameters to be passed to the main view template of OceanTEA to, e.g., load predefined libraries. The following parameters are available:

- `libD3js: true`
- `libCanvasPlot: true`
- `libFontAwesome: true`
- `libBootstrapJS: true`
- `libBootstrapSlider: true`

By default, jQuery⁵ and the Bootstrap⁶ CSS are loaded.

³<https://github.com/a-johanson/oceantea-visualization-gateway/blob/master/plugins.js>

⁴For a full working example, see:

<https://github.com/a-johanson/oceantea-visualization-gateway/blob/master/plugins.js#L10>

⁵<https://jquery.com>

⁶<https://getbootstrap.com>