

How To: Add API Composites to the Ed-Fi ODS / API Solution

This section outlines the steps necessary to integrate and activate the Ed-Fi Composite definitions for use in an Ed-Fi ODS / API. It is assumed that the Ed-Fi ODS / API has been successfully downloaded and is running in a local environment per the instructions in the [Getting Started](#) documentation.

The steps in Visual Studio can be summarized as:

- [Step 1. Create the Composites Project](#)
- [Step 2. Integrate Composites into the Solution](#)
- [Step 3. Verify Changes](#)

Each step is outlined in detail, below.

Downloads

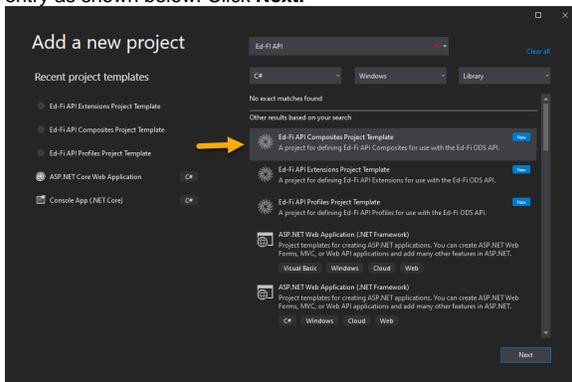
The following GitHub link contains source files for the Composite described in this article:

[Composite Source Files](#)

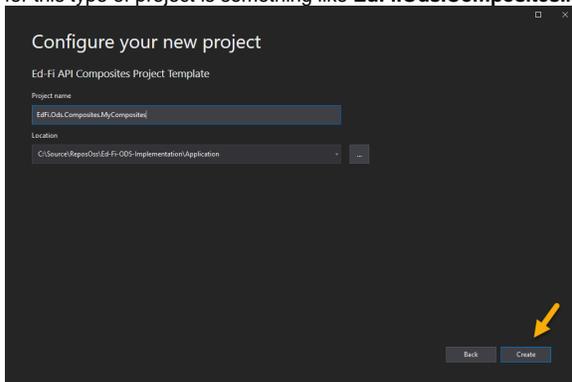
Step 1. Create the Composites Project

Step 1.1. Add a Composite Project Using the Visual Studio Project Template. Visual Studio Project Template can be installed by following the steps in the [Project Templates Installation](#) section of this documentation.

- To add a project to your Ed-Fi-Ods Visual Studio Solution, **right-click** on the "Composites" folder. Select **Add > New Project**.
- In the "Add New Project" dialog, find and select the "Ed-Fi API Composites Project Template" entry as shown below. Click **Next**.



- Enter the project name for the new project and click **Create**. The suggested naming convention for this type of project is something like **EdFi.Ods.Composites.MyComposites**.



Step 1.2. Update the Marker Interface file.

To integrate the Composite with the API, start by ensuring you have a marker interface in the root of your Composites project.

This interface is an idiom used in the Ed-Fi Visual Studio Solution to enable a strongly typed mechanism for obtaining a reference to the .NET assembly. If you used the Visual Studio Project template to create your composite, a file will already exist — but you'll need to rename the interface and the file to match the convention (e.g., **Marker_EdFi_Ods_Composites_MyComposites.cs**). The marker interface file should have the following code:

```

namespace EdFi.Ods.Composites.MyComposites
{
    public interface Marker_EdFi_Ods_Composites_MyComposites { }
}

```

Step 1.3. Update the **Composites.xml** file to add the appropriate composite definition.

The Visual Studio Project Template creates a sample **Composites.xml** file. Consult [API Composite Resources](#) for guidance. The Composite Definition should look something like the following:

```

<?xml version="1.0" encoding="utf-8"?>
<CompositeMetadata organizationCode="ed-fi">
  <Category displayName="My Sample Composites" name="MyComposite">
    <Routes>
      <Route relativeRouteTemplate="/sections/{Section.Id}/
{compositeName}" />
    </Routes>
    <Composites>
      <Composite name="Student">
        <Specification>
          <Parameter name="Section.Id" filterPath="
StudentSectionAssociation->Section.Id" />
        </Specification>
        <BaseResource name="Student">
          <Property name="StudentUniqueId" />
          <Property name="FirstName" />
          <Property name="LastSurname" />
          <Property name="BirthDate" />
        </BaseResource>
      </Composite>
    </Composites>
  </Category>
</CompositeMetadata>

```

 Composites cannot include resource model members added via Ed-Fi Extensions. ODS / API composite definitions currently support only the resource models from the Ed-Fi Core.

The `CompositeMetadata` attribute `organizationCode` is a required attribute and indicates the Organization to which the composite belongs. This value is carried from the XML definition all the way into the API route definitions. The `organizationCode` in combination with the Category name are used to identify the composite being requested from the API.

Step 1.4. Save the Project.

Step 2. Integrate Composites into the Solution

To integrate the Composite Resources into the solution, add a reference to new Composites project you constructed in the previous step in the **EdFi.Ods.WebApi** project (located in the "Entry Points" folder).

Step 3. Verify Changes

Save all modified files, then run the application and view the Ed-Fi ODS / API using Swagger. The following new API Composite resource should be available:

 You may have to clear your browser cache in order for Swagger UI to display the new resource.

ed-fi

API Section: Composites: MyComposite

Ed-Fi Operational Data Store API

Base URL: `localhost:54746/api/odds/`
<https://github.com/ed-fi/operational-data-store/blob/master/docs/2016-01-composites/my-composite-overview.md>

The Ed-Fi ODS API enables applications to read and write education data stored in an Ed-Fi ODS through a secure REST interface. The Ed-Fi ODS API supports both transactional and bulk modes of operation.

Authorize

students

- GET /ed-fi/composites/MyComposite/Students. Retrieves specific resources using the resource's property values using the "Get" pattern.
- GET /ed-fi/composites/MyComposite/Students/{id}. Retrieves a specific resource using the resource's identifier using the "Get By ID" pattern.
- GET /ed-fi/composites/MyComposite/sections/{Section_id}/students. Retrieves specific resources using the resource's property values using the "Get" pattern.

The organizationCode mentioned in Step 1.3 ("ed-fi" in this example) can be seen in the Composite URLs above (e.g., /ed-fi/composites/MyComposite/Students). To successfully retrieve a composite, this value (defined in the XML definition) must be present as the first segment of a composite URL.