

## **Wikiprint Book**

**Title: Sample code**

**Subject: Ecopath Developer Site - EcosimVariableTimestepPluginExample**

**Version: 13**

**Date: 2019-07-17 16:18:09**

## Table of Contents

<b>Sample code</b>	<b>3</b>
<b>Running Ecosim on a variable time step</b>	<b>3</b>
Sub time step data	3
Sub time step plugin points	3
<b>Running Ecosim on a separate thread</b>	<b>3</b>

## Sample code

Source code for a sample plugin can be found at

...Ecopath6\Sources\EwETutorials\CEFASPluginSample

To view and run the sample code in Visual Studio

- i. Load ...Ecopath6\Sources\EwE6.sln into Visual Studio.
- ii. Add the CEFASPluginSample.vbproj by right clicking on **Solution Explorer** and selecting **Add>Existing Project...** and finding ...Ecopath6\Sources\EwETutorials\CEFASPluginSample
- iii. Set a reference to CEFASPluginSample in the ScientificInterface project by right clicking on the ScientificInterface node in the **Solution Explorer** and selecting **Add Reference...** click the **Add...** button from the **Projects** tab select CEFASPluginSample
- iv. Run the CEFASPluginSample by compiling and running EwE6. From EwE6 main menu select **Tools>CEFAS plug-in sample**

## Running Ecosim on a variable time step

Ecosim can be run on a variable time step via a plugin. When run from the Scientific Interface Ecosim runs 1 time step per month, 12 time steps per year. The number of timesteps per month can be set in code via the `cEcosimDataStructures.StepsPerMonth` property. This allows a plugin to run Ecosim on sub time steps, multiple time steps in a month.

### Sub time step data

When running on sub time steps Ecosim will only update its results objects `cCore.cEcoSimResults()`, `cCore.cEcosimGroupOutput()`, `cCore.cEcosimOutput()` and `cCore.cEcosimFleetOutput()` once at the end of the month. Data computed for the sub time step can only be access via the `cEcosimDataStructres` provided by the `IEcosimSubTimestepsPlugin.EcosimSubTimeStepBegin()` and `IEcosimSubTimestepsPlugin.EcosimSubTimeStepEnd()` plugin points. The `IEcosimBeginTimestepPlugin.EcosimBeginTimeStep()` and `IEcosimEndTimestepPlugin.EcosimEndTimeStep()` will only be call at the start and end of the month not during the sub time step and will not contain the sub time step data.

### Sub time step plugin points

`IEcosimSubTimestepsPlugin.EcosimSubTimeStepBegin(ByRef BiomassAtTimestep() As Single, ByVal TimeInYears As Single, ByVal DeltaT As Single, ByVal SubTimestepIndex As Integer, ByVal EcosimDatastructures As Object)`

- Called at the start of a sub time step.
- `BiomassAtTimestep()` contains the biomass that will be used for the time step. Any changes to biomass by a plugin point will be use be by Ecosim for the time step.

`IEcosimSubTimestepsPlugin.EcosimSubTimeStepEnd(ByRef BiomassAtTimestep() As Single, ByVal TimeInYears As Single, ByVal DeltaT As Single, ByVal SubTimestepIndex As Integer, ByVal EcosimDatastructures As Object)`

- Called at the end of a sub time step.
- `BiomassAtTimestep()` contains the biomass computed for the time step.

## Running Ecosim on a separate thread

Ecosim can be run on a separate thread from the user interface by setting the `cEcosimDataStructures.bMultiThreaded` Boolean flag to True. The next time Ecosim is run it will run on its own thread. Once the Ecosim run has completed the `cEcosimDataStructures.bMultiThreaded` flag will be automatically set to its default value of False and all subsequent run will be on the main application thread. All plug-in calls that originate on a thread other than the main thread will be marshaled to the main thread. This is for robustness, in .NET user interface controls cannot be accessed from a thread other than that which they were created on. Plug-ins are created and run on the main thread by the EwE framework. By automatically marshaling all plug-in calls to the main thread plug-in points do not have to marshal calls to an interface.