Monitoring a Custom API-Based Events Source

This topic explains how to configure monitoring for a custom API-Based Events source.

To configure the Event Monitor tool to monitor a custom API-based event source:

1. Double-click the Event Monitor tool to open up the tool configuration panel.
2. In the **Event Source** tab, select **Custom API-Based Events Source** as the platform, then configure the following options:
    a. In the **Connection** area, specify your connection settings.
    b. In the **Event Retrieval** area, specify the event retrieval pattern you want to use (polling at a specified time interval, polling after each test execution, or subscribing to an event producer).
    c. In the **User Code** area, specify the location of your custom event monitoring application or scripts. See Extensibility API Patterns for details.
3. In the **Options** tab, modify settings as needed.
    - **Clear the event viewer before each event monitor run** determines whether SOAtest automatically clears the Event Monitor event view (both text and graphical) whenever Event Monitor starts monitoring.
    - **Include test execution events in the XML event output** specifies whether the Event Viewer tab and XML output display show only the monitored messages and events, or if they also indicate when each test started and completed. Enabling this option is helpful if you have multiple tests in the test suite and you want to better identify the events and correlate them to your test executions.
    - **Wrap monitored messages with CDATA to ensure well-formedness of the XML event output** should be disabled if you expect the monitored events' message con-tent to be well-formed XML. This will make the messages inside the events accessible via XPaths, allowing the message contents to be extracted by XML Transformer or validated with XML Assertor tools.
        - If the message contents are not necessarily XML, you should enable this option to ensure that the XML output of the Event Monitor tool (i.e. the XML Event Output for chaining tools to the Event Monitor, not what is shown under the Event Viewer) is well-formed XML by escaping all the message contents. This will make the content of these messages inaccessible by XPath since the message technically becomes just string content for the parent element.
        - Note that the Diff tool's XML mode supports string content that is XML. In other words, no matter which option you select here, the Diff tool will still be able to diff the messages as XML, including the ability to use XPaths for ignoring values.
    - **Maximum monitor execution duration** specifies the point at which the test should timeout—in case another test in the test suite hangs, or if no other tests are being run (e.g., if you execute the Event Monitor test apart from the test suite, then use a custom application to send messages to system).
    - **Event polling delay after each test finishes execution (milliseconds)** specifies how long Event Monitor waits between the time the test ends and the time it retrieves the events.