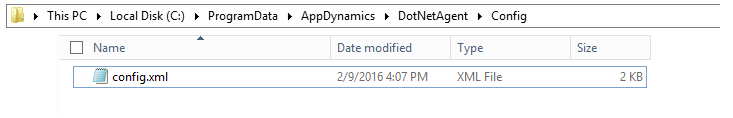
# **Installation and Configuration**

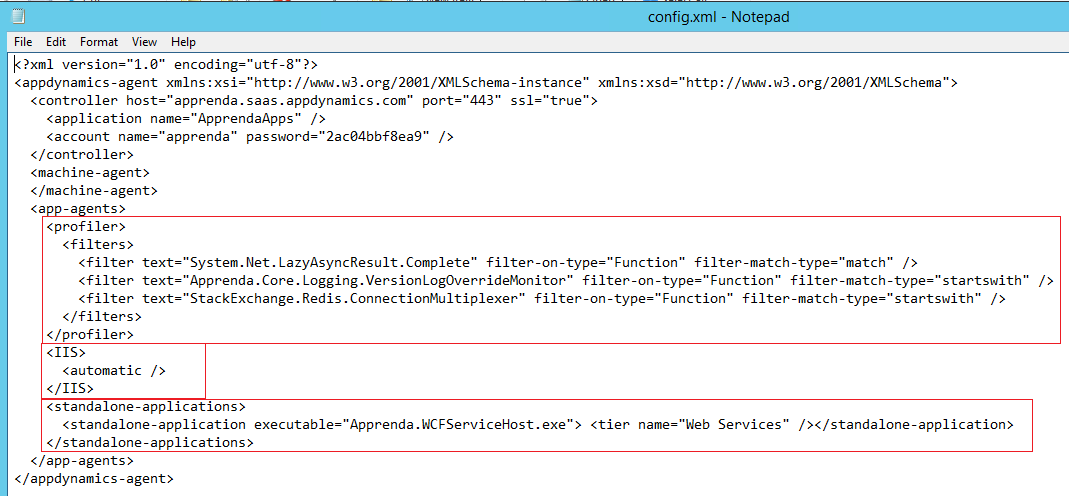
In this guide, we will walk you through a few simple steps to integrate Apprenda with AppDynamics, a leader in APM software.

## Installing and Modifying AppDynamics

1. Install the AppDynamics .NET monitoring agent (dotNetAgentSetup.msi) on every Apprenda node that is part of the compute grid for Apprenda PaaS applications
2. Install the AppDynamics database monitoring agent on all the database nodes that are part of the Apprenda grid
3. Optionally install the AppDynamics server monitoring agent on the compute and database nodes
4. Connect the AppDynamics agents to the controller for the AppDynamics installation you will use (either the hosted AppDynamics controller or the SaaS offering from AppDynamics)
5. On every Apprenda compute node where the agent is installed, modify the config.xml file from AppDynamics. The config.xml file is by default located at C:\ProgramData\AppDynamics\DotNetAgent\Config



1. In the config.xml file, enable the instrumentation of all IIS application pools and also instruct AppDynamics to monitor the WCF hosting container executable for all Apprenda WCF services, Apprenda.WCFServiceHost.exe. Any application monitored by the two above modifications will land under the uber umbrella of a single AppDynamics application as indicated by the config file. In this example, it is called ApprendaApps.
2. Make the necessary AppDynamics profile changes to light up Apprenda application monitoring for IIS-based applications
3. Example of the changes necessary in the config.xml are highlighted below and also a sample config.xml is included in the GitHub integration page under the folder AppDynamicsSampleConfigFile

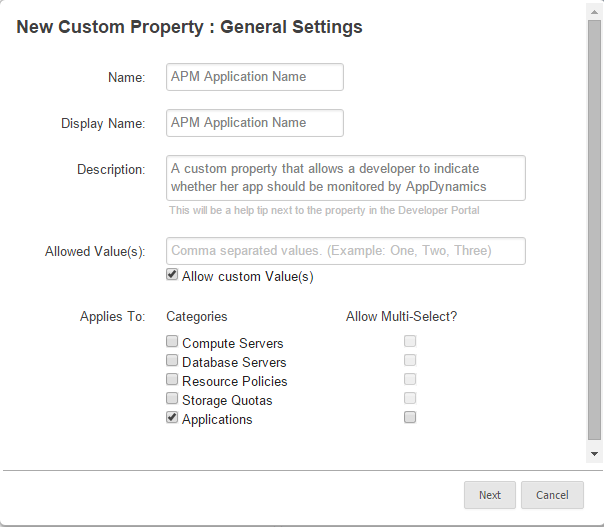


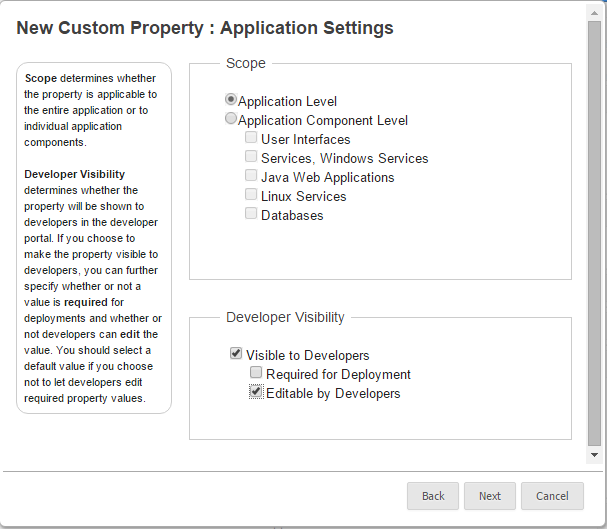
1. After the necessary modifications are made to config.xml, reboot the windows nodes for all the changes to take effect.

## Creating the Custom Property

Now switch to the Apprenda Operator Portal (also known as SOC) to create the Apprenda configuration for this integration.

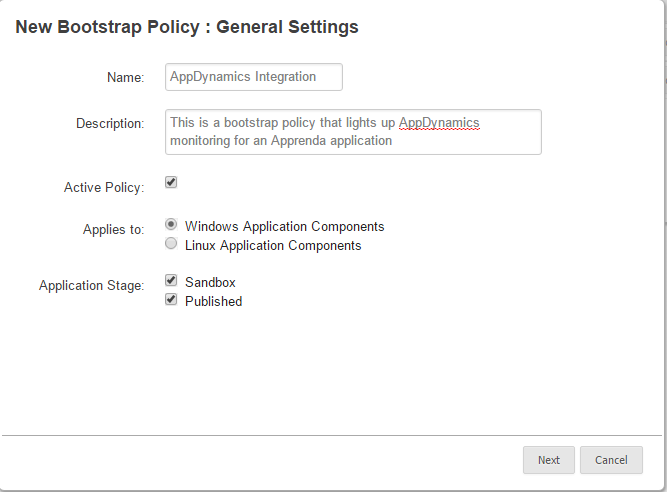
1. Create a Custom Property called “APM Application Name”. Each Apprenda application that wants to enable APM should fill in that custom property. The custom property should apply to Applications, allows a custom value to be set, and is visible and editable by developers.

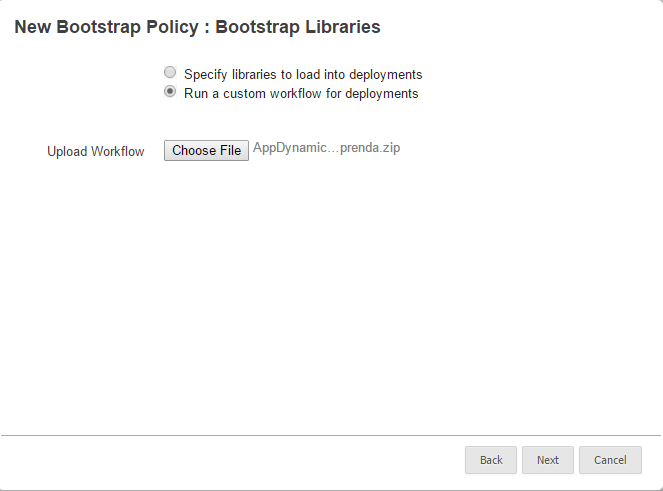


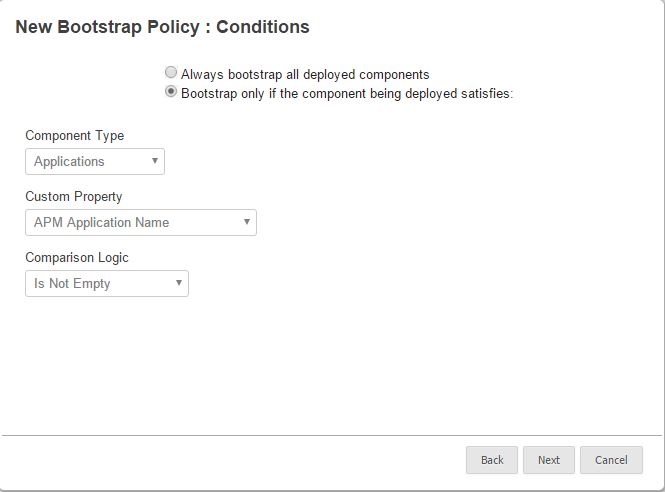


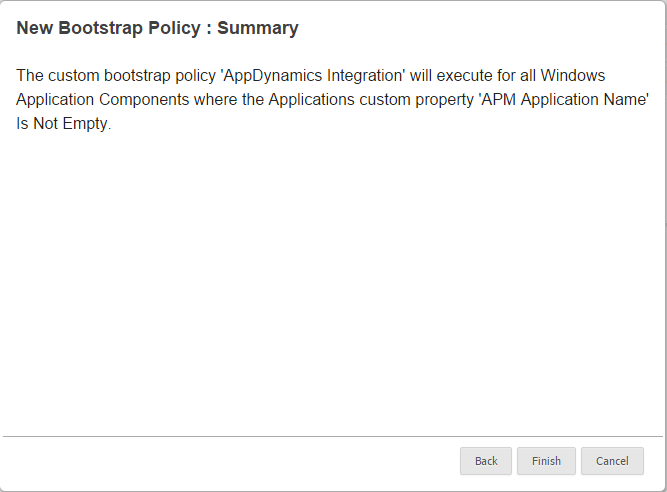
## Creating the Bootstrap Policy

1. Create the Bootstrap Policy that will trigger the workflow for enabling AppDynamics integration for Apprenda applications that indicate that APM monitoring is needed. Apprenda applications use the “APM Application Name” custom property to trigger the bootstrap policy workflow. The Bootstrap Policy workflow binaries (AppDynamicsBSP\_UploadToApprenda.zip) are included in the GitHub integration page under the folder BootstrapperBinariesandFolderStructure









# **Usage**

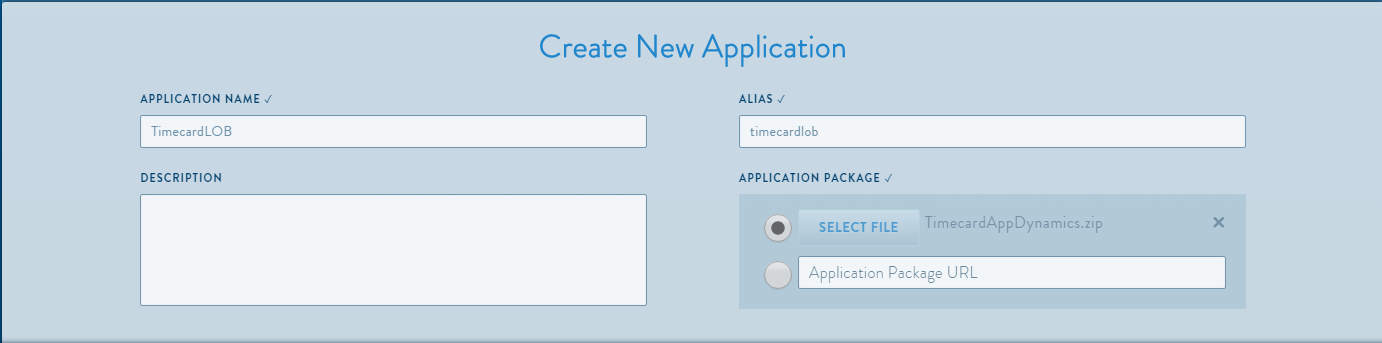
Having set up AppDynamics and the proper policies in Apprenda, it is time to demonstrate how an Apprenda application can be instrumented for APM monitoring.

## Creating an Application

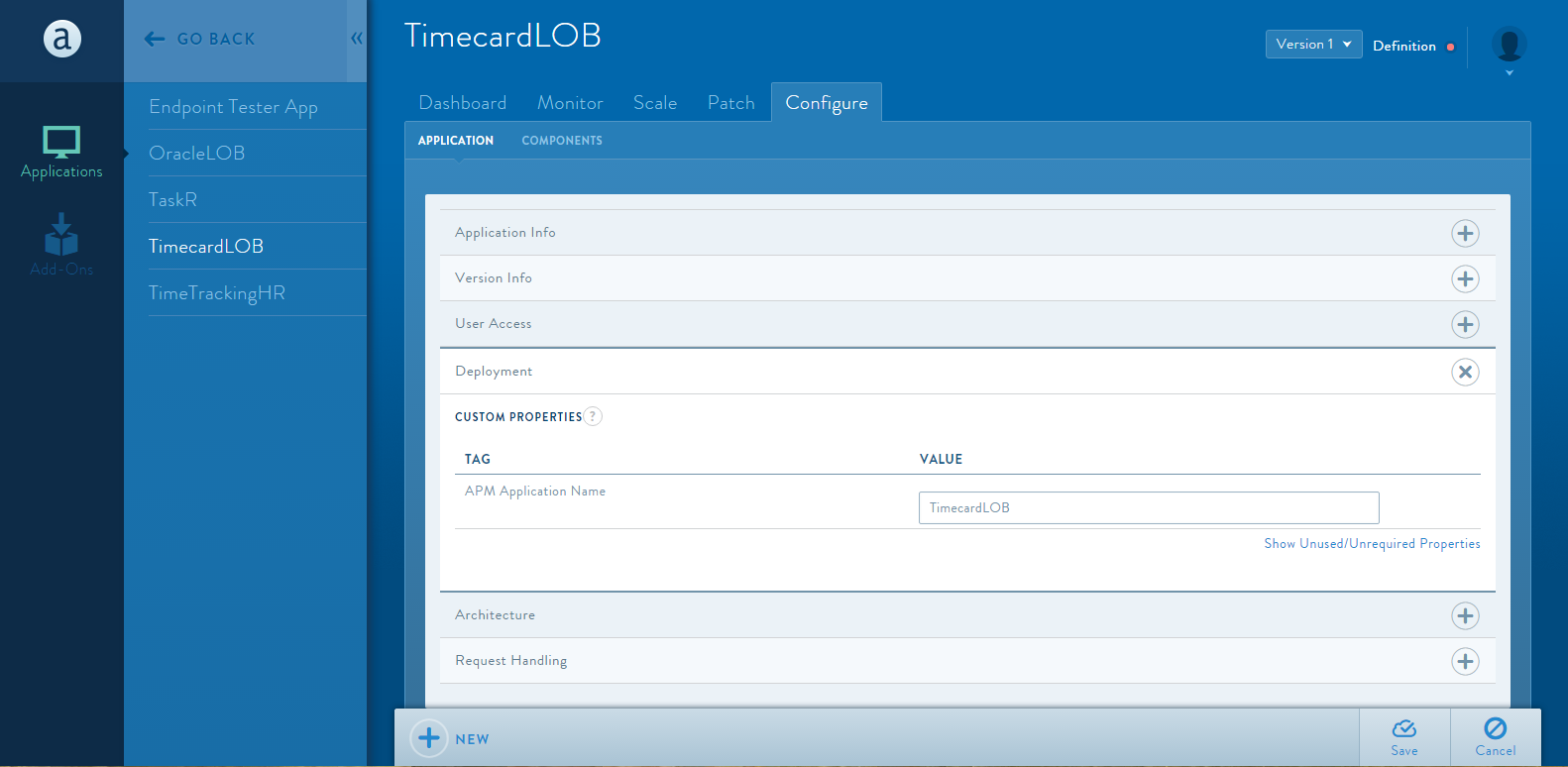
1. Modify your Apprenda application using the DeploymentManifest.xml to indicate the Tier and Application names. These names will be reflected as high-level applications in the AppDynamics portal. You can control the tier name for your WCF web services through the manifest. The tier name for IIS-based apps is hardcoded in the Bootstrap policy workflow.
2. The application custom property “APM Application Name” should be filled in as this is the custom property that will trigger the execution of the AppDynamics Bootstrap policy
3. The sample manifest and Apprenda application package TimecardAppDynamics.zip is included in the SampleApprendaApplication folder of the GitHub integration page.



1. Create the new application in the Apprenda Developer Portal



1. Verify the APM Application Name custom proprety is set in the application’s configuration page



1. Launch the application and use it just like you would use any other Apprenda application
2. AppDynamics will immediately instrument your application and start sending data back to the AppDynamics controller
3. Visit the AppDynamics portal, find your application and use the AppDynamics Application Intelligence capabilities to monitor the application

