

# Working with a Remote Dedicated SOAtest Server

This section describes the available SOAtest server implementations and how to interact with a remote SOAtest and Virtualize server from the desktop.

- [Introduction](#)
- [Adding a Remote Server to the Desktop View](#)
- [Transferring Files Between the Remote Server and the Local Machine](#)

## Introduction

A common SOAtest infrastructure configuration includes a SOAtest server deployed to the team build machine and SOAtest desktop installations for developers/testers. If your license permits, you can also run a Virtualize server and Virtualize desktops, which enable you to quickly create and deploy shareable, reusable virtual environments. By adding Continuous Testing Platform (CTP) to your infrastructure, you can enable browser access to test assets, virtual environments, test data, and other components to facilitate testing practices.

The crux of the infrastructure is the SOAtest (and/or Virtualize) server. You can execute SOAtest functionality on remote servers using either the [SOAtest Web Services Interface](#) or [SOAtest and Virtualize Server](#) implementation. In both implementations, the remote SOAtest server and desktop instance interacting with it must be the same version to ensure interoperability. Sub-minor version difference, such as service pack updates, however, do not impact interoperability.

## SOAtest Web Services Interface

You can install SOAtest on a remote machine and access the web services API to execute SOAtest functionality. This enables you to automate test execution as part of your continuous integration infrastructure (i.e., nightly build). You can also access the SOAtest server from the [SOAtest Server View](#) in a desktop implementation.

Details on how to set up and run SOAtest from a command line, locally or on a remote server, are in the [Testing from the Web Service Interface](#) section.

## SOAtest and Virtualize Server

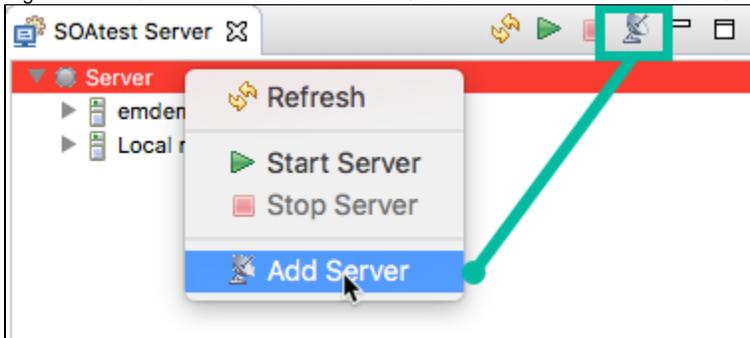
The dedicated SOAtest and Virtualize server is a lightweight implementation for executing tests created in the SOAtest desktop or CTP, as well as hosting service virtualization assets created with Virtualize and/or CTP. The server architecture is designed to provide a smaller memory footprint for higher performance of individual servers, as well as to let organizations take advantage of standardized application deployment infrastructures for availability, clustering, and scalability. It is designed to be deployed within containers such as Docker.

See the [Deploying the SOAtest and Virtualize Server WAR File](#) section for instructions on how to deploy and configure the server.

## Adding a Remote Server to the Desktop View

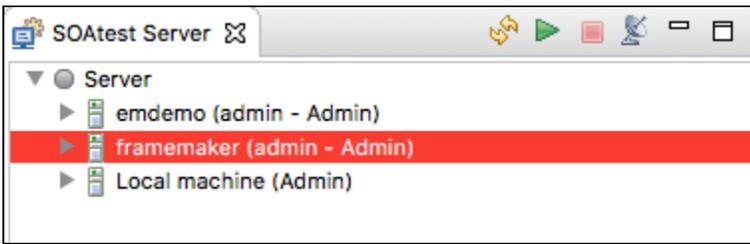
The SOAtest Server view is the primary interface for deploying test assets to the remote server (also see [SOAtest Server View](#)).

1. If the SOAtest Server view is not already open, Choose **Window> Show View> SOAtest Server**.
2. Right-click the **Server** node and choose **Add Server**. You can also click the **Add Server** button (SOAtest icon).



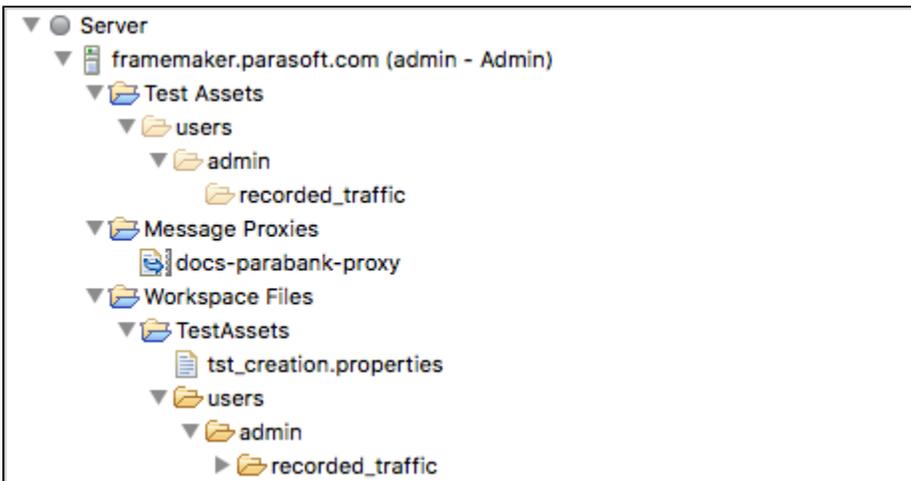
3. Specify the server connection settings when prompted:
  - Protocol: HTTP or HTTPS
  - Host: Name of the host where the server is deployed
  - Display Name: Name to appear in the SOAtest Server view
  - Port: Port on the host where the server is deployed

The server will be added to the view.



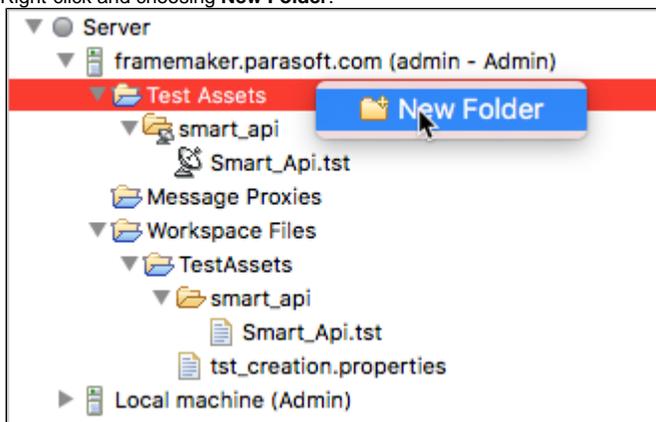
The SOAtest Server view shows the Parasoft artifacts deployed to the remote server. The artifacts are grouped into the following folders:

- **Test Assets:** This folder shows the .tst files on the server. You can drag files from your local Test Case Explorer view into the Test Assets folder and execute them through the REST API.
- **Message Proxies:** If message proxies are deployed to your server, they will appear in this folder. Message proxies are Parasoft Virtualize assets that enable you to execute tests against virtualized services.
- **Workspace Files:** This folder shows additional files stored in the remote server's workspace. When a test asset is deployed to a remote server, the file(s) is written to the TestAssets subfolder.



You can add subfolders to the Test Assets folder to keep your work organized.

1. Right-click and choosing **New Folder**.



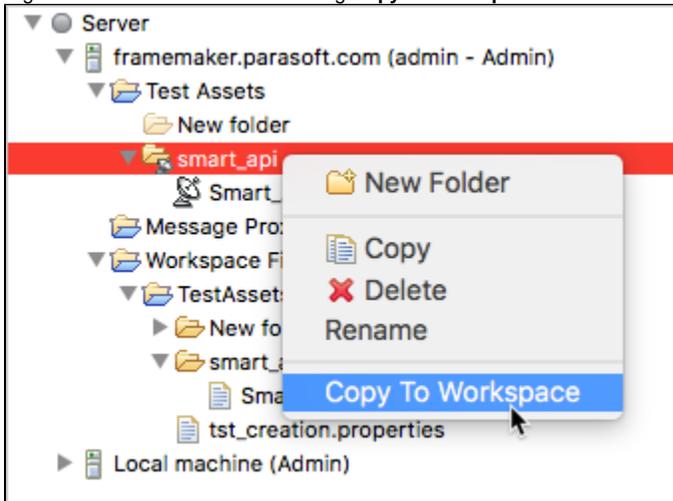
2. Specify a name when prompted and click **OK**.

## Transferring Files Between the Remote Server and the Local Machine

To copy assets from the local instance of SOAtest, drag files from the Test Case Explorer view to the Test Assets folder or a subfolder in the SOAtest Server view.

You can also copy files and folders from the remote server to your local SOAtest:

1. Right-click a file or folder and choosing **Copy to Workspace**.



2. Specify the parent folder to copy the remote asset to when prompted and click **OK**.