PC-DOCTOR, INC.
Delivering Confident Computing™

PC-Doctor 5 Test Scripts

Guidelines for Creating and Modifying PC-Doctor 5 Test Scripts

Last revised: 2005 - 4 - 5

Contents

Modifying Test Scripts Manually

All PC-Doctor product test scripts are in .xml format. There are a wide degree of test script settings you can modify in the test script. Below is a basic diagram for PC-Doctor script composition.

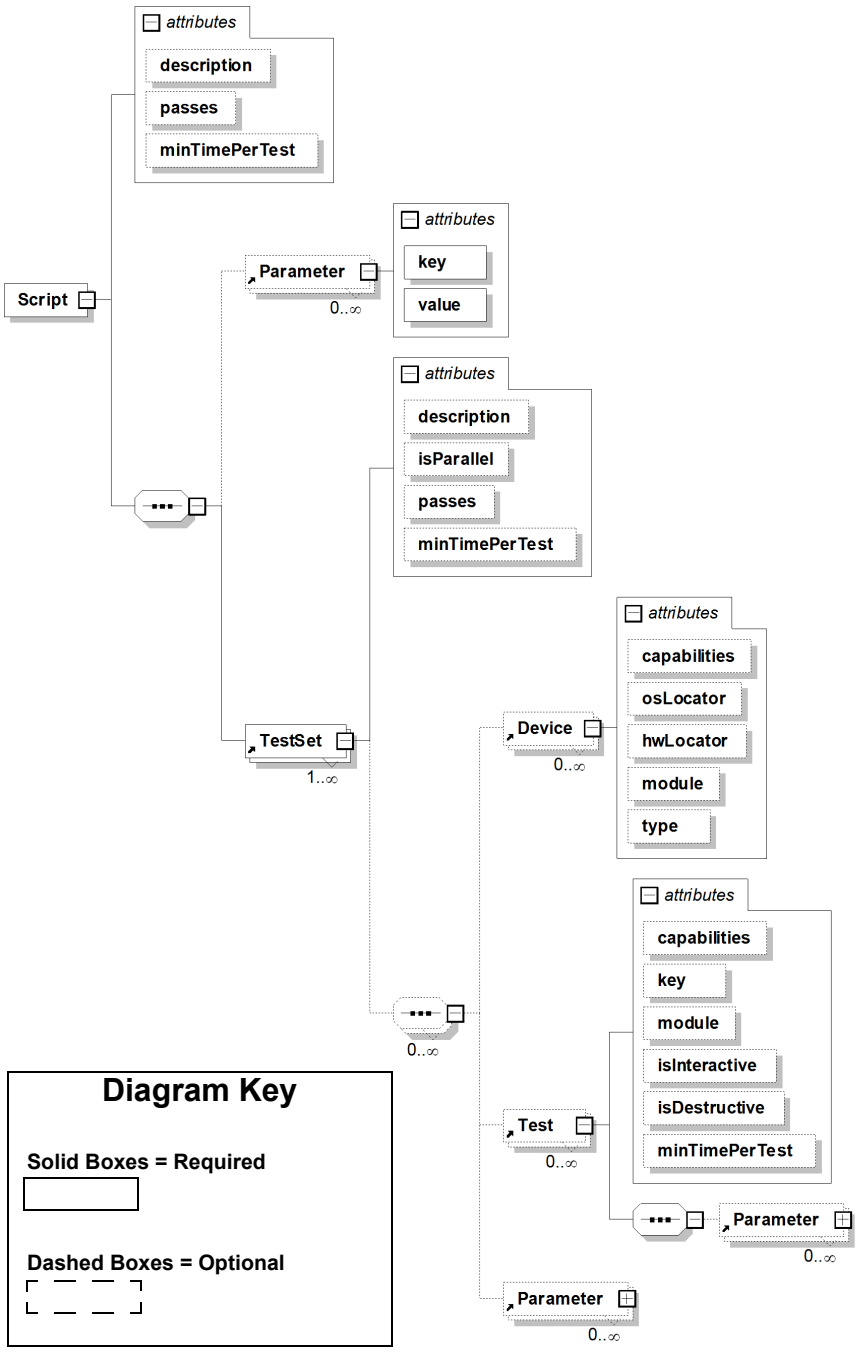

Figure 1: PC-Doctor Test Script Composition Elements

The <Script> Element

PC-Doctor test scripts consist of elements listed in a hierarchal relationship. The root element is <Script>. You can apply the following attributes to the <Script> element:

- **description**: A description of the script. Apply this attribute to the <Script> element when referencing the script in logs or other places.
- **passes**: The number of times to run the script.
- **minTimePerTest**: Runs a script repeatedly until the specified minimum time elapses.

The <TestSet> Element

Nested within the <Script> element are one or more <TestSet> elements. A <TestSet> contains any number of <Device> and <Test> elements. When PC-Doctor loads a test script, the <Device> and <Tests> within a particular <TestSet> are paired together to form test/device pairs. PC-Doctor products will warn you if it fails to find a test for specified devices, or when it fails to find a device for specified tests.

PC-Doctor products will automatically populate a <TestSet> element with all available tests for a device if you do not specify any test elements. If you do not specify any <Device> elements, PC-Doctor products will automatically populate the <TestSet> element with all available devices that the specified test supports.

You can apply the following attributes to the <TestSet> element:

- **description**: A description of the test set. Apply this attribute to the <TestSet> element as a reference to the test set when PC-Doctor displays error or warning messages.
- **isParallel**: Specifies tests in the test set to run in parallel with each other and in parallel with other test sets specified to run in parallel. This attribute supports values of “true” and “false”, and is set to “false” by default.
- **passes**: The number of times to run tests within a test set.
- **minTimePerTest**: Runs a test repeatedly until the specified minimum test time elapses.

Test Matching

<Test> elements contain attributes to restrict which tests in the system that they will match:

- **capabilities**: A test must support a specified capability to match up with the <Test> element. To list the available capabilities, launch the PC-Doctor Command Line Interface and run the **pcd capabilities** command.

- **key**: The key of the test. Multiple device types can have tests with the same key, so you may need to combine the key attribute with a module or capabilities attribute.

- **module**: A test must belong to the specified module to match up with the <Test> element.

- **isInteractive**: If the value for this attribute is “true”, a test will match if it requires interactivity. For example, Interactive Keyboard Test, Interactive Mouse Test, and so on. if the value for this attribute is “false”, a test will match if it is not interactive. if you do not specify a value for this attribute, then test matching will not be restricted by interactivity. This attribute supports values of “true” and “false”, and is left unspecified by default.

- **isDestructive**: If the value for this attribute is “true”, a test will match if it is considered destructive. For example, the hard drive pattern test actually writes test data to the hard drive, possibly overwriting existing data. For this reason, the Hard Drive Patterns Test is considered destructive. if the value for this attribute is “false”, a test will match if it is not considered destructive. if you do not specify a value for this attribute, then PC-Doctor will not limit test matching to destructive/non-destructive tests.

- **minTimePerTest**: Runs a test repeatedly until the specified minimum test time elapses.

Device Matching

<Device> elements contain attributes to restrict which devices in the system that they will match:

- **type**: A device must belong to a specified device type group to match up with the <Device> element.

- **capabilities**: A device must support a specified capability to match up with the <Device> element.

- **osLocator**: To match, a device must have this operating system locator. This locator is dependent on the operating system. For example, the primary hard drive in a Linux system might have an osLocator of "/dev/hda", while in Windows it might be "\\.\c:".

- **hwLocator**: To match, a device must have this hardware locator. Using this is the easiest way to test only one device type when multiple device types are present in the system. For example, if there are two CPU's, you can restrict the match to only the first one by using *osLocator="CPU:0"*.

- **module**: To match, a device must come from this system information module.

Test Script Parameters

Parameters are values that change the way tests run. You can place parameters anywhere in a script. At the top level (with <TestSets>), they apply to tests within all of the test sets. A parameter within a <TestSet> element applies to all the tests in that test set. When placed under a <Test> element, only that <Test> element is affected by the parameter. If a parameter is set at multiple levels, the parameter values at deeper levels (for example, under a <Test> element) will override parameter values at higher levels (for example, under the <Script> element). The parameters for tests are found in associated p5i files. For example, parameters for CPU tests are found in the pcdrcpu.p5i file.

You can apply the following attributes to the <TestSet> element:

- **key**: the key of the parameter.
- **value**: the value of the parameter.

Modifying Test Scripts to Conduct Custom Testing

For example, if you wanted to test COM1 ports but not test COM2 ports, you would need to add a <Device> element along with the <Test> element so the script matches the specified device. You manually specify this in the test script by doing the following:

1. Use System Information to determine the hardware locators for COM1 and COM2. Generally, the hardware locator for COM1 is COM:0.
2. Open the test script with any text editor. PC-Doctor, Inc. recommends using a text editor with syntax highlighting.
3. To run all available tests for COM1 ports, add the following <Device> element nested within a <TestSet> element:

```
<TestSet>
  <Device hwLocator="COM:0"/>
</TestSet>
```

4. To modify what tests run on COM1 as part of the script, include the following <Test key> elements nested within the <TestSet> element:

```
<TestSet>
  <Device hwlocator="COM:0"/>
  <Test key="InternalRegisterTest"/>
  <Test key="ControlSignalsTest"/>
  <Test key="DataSendandReceiveTest"/>
</TestSet>
```

   The above example specifies PC-Doctor to run the Internal Register, Control Signals, and Data Send and Receive tests on the COM1 port.

Sample Test Scripts

The following are sample PC-Doctor test scripts. Below is a sample test script called “Burn-In Test” configured to perform burn-in testing on the CPU, memory, hard drive, serial port, and parallel port:

```
<Script description="Burn-In Test" minTimePerTest="60">
        <TestSet isParallel="true">
                <Device type="CPU"/>
                <Test key="CacheTest"/>
        </TestSet>
        <TestSet isParallel="true">
                <Device type="Memory"/>
                <Test key="AdvancedPatternTest"/>
        </TestSet>
        <TestSet isParallel="true">
                <Device type="HardDrive"/>
                <Test key="RandomSeekTest"/>
        </TestSet>
        <TestSet isParallel="true">
                <Device type="SerialPort"/>
                <Test key="InternalRegisterTest"/>
        </TestSet>
        <TestSet isParallel="true">
                <Device type="ParallelPort"/>
                <Test key="InternalTest"/>
        </TestSet>
</Script>
```

Below is a sample test script called “System Test” configured to test the hard drive, floppy drive, optical drives, and CPU:

```
<Script description="System Test">
        <TestSet>
                <Device type="HardDrive"/>
                <Test key="RandomSeekTest"/>
        </TestSet>
        <TestSet>
                <Device type="FloppyDrive"/>
                <Test key="PatternTest"/>
                <Parameter key="PromptUserForMedia" value="false"/>
        </TestSet>
        <TestSet>
                <Device capabilities="Optical"/>
                <Test key="LinearSeekTest"/>
                <Test key="RandomSeekTest"/>
                <Parameter key="PromptUserForMedia" value="false"/>
        </TestSet>
        <TestSet>
                <Device type="CPU"/>
                <Test key="RegisterTest"/>
                <Test key="MathRegisterTest"/>
```

```
            <Test key="CacheTest"/>
            <Test key="MMXTest"/>
        </TestSet>
</Script>
```

Below is a sample test script called “Matching Demonstration” configured for device matching:

```
<Script description="Matching Demonstration" passes="2" >
<Parameter key="NumberOfSeeks" value="2000"/>
        <!-- the following two testsets should be the same -->
        <TestSet description="All devices that can read CDROMs">
            <Device capabilities="CDROM"/>
        </TestSet>
        <TestSet description="All tests that can test CDROMs">
            <Test capabilities="CDROM"/>
            <Parameter key="NumberOfSeeks" value="1000"/>
        </TestSet>
        <!-- test overlapping devices -->
        <TestSet description="overlapping devices">
            <Device type="CPU"/>
            <Device type="CPU"/>
        </TestSet>
        <!-- test overlapping tests -->
        <TestSet description="overlapping tests">
            <Test key="RandomSectorReadTest"/>
            <Test key="RandomSectorReadTest"/>
        </TestSet>
        <!-- test test/device mix -->
        <TestSet passes="2" description="test/device mix">
            <Device type="CPU" />
            <Device type="HardDrive"/>
            <Test key="RegisterTest"/>
            <Test key="RandomSectorReadTest">
                <Parameter key="NumberOfSeeks" value="1000"/>
            </Test>
        </TestSet>
        <!-- device matching by various test properties -->
        <TestSet isParallel="false" description="Device-centric Test Set">
            <Device osLocator="IDE0"/>
            <Device type="CDROM" required="true"/>
            <Device module="pcdrcpu"/>
        </TestSet>
        <!-- matching by various test properties -->
        <TestSet isParallel="true" description="Test Set defined by Tests">
            <Test isInteractive="true"/>
            <Test isDestructive="true"/>
        </TestSet>
</Script>
```

For More Information

Contact

PC-Doctor, Inc.
9805 Double R Blvd. Suite 301
Reno, NV 89521

Phone: 775-336-4000
Fax: 775-336-4099
E-mail: support@pc-doctor.com
Web: http://www.pc-doctor.com

