This document describes

* what OPBM is
* how to navigate the OPBM windows
* what actions are available on each window
* how to use the 3 interfaces: simplified run interface, developer interface, command line interface.

Use the hyperlinks below to go to the specific OPBM documentation topic.

[Introducing OPBM](#Introducing_OPBM)

[Advantages of OPBM](#Advantages_of_OPBM)

[Terminology](#Terminology)

[Scoring Methodology](#Scoring_methodology)

[Minimum Requirements to Run OPBM](#Minimum_requirements_to_run_OPBM)

[How to Set-up and Install OPBM](#How_to_set_up_and_install_OPBM)

[Starting OPBM](#Starting_OPBM)

[Parts of the OPBM – Office Productivity Benchmark Main Window](#Parts_of_the_Main_Window)

[The HUD (Heads Up Display)](#HUD_heads_up_display)

[The Results Viewer](#Results_viewer)

[The Simplified Run Interface](#Simplified_Run_Interface)

[About the Simplified Run Interface](#About_the_simplified_run_interface)

[The Simplified Run Interface Main Window](#Simplified_run_interface_main_window)

[Using the **Trial Run** button](#Using_the_trial_run_button)

[Using the **Official Run** button](#Using_the_official_run_button)

[Using the **View Previous Results** button](#Using_the_view_previous_results_button)

[Using the **Developer Interface** button](#Using_the_developer_interface_button)

[The Developer Interface](#Developer_Interface)

[About the Developer Interface](#About_the_developer_interface)

[The Run Modules: Atoms](#Run_modules_atoms)

[The Run Modules: Molecules](#Run_modules_molecules)

[The Run Modules: Scenarios](#Run_modules_scenarios)

[The Run Modules: Suites](#Run_modules_suites)

Full Benchmarks: Trial Run

Full Benchmarks: Official Run

Shortcuts:

Maintenance: Script Definition

Maintenance: UI Settings

[The Command Line Interface](#Command_Line_Interface)

[About The Command Line Interface](#About_the_command_line_interface)

[Command Line Syntax](#Command_line_syntax)

[Running OPBM from the command line](#Running_OPBM_from_the_command_line)

[Executing scripts from the command line](#Executing_scripts_from_the_command_line)

[Benchmarking across multiple machines](#Benchmarking_across_multiple_machines)

[OPBM Harness Configuration](#OPBM_harness_configuration)

Problems [Running](#Problems_running_opbm) OPBM

[Auto\_logon\_disable](#Auto_logon_disable)

[Conflicts and Resolutions window](#Conflicts_and_resolutions)

[Getting Help](#Getting_help)

**Introducing OPBM**

OPBM (Office Productivity Benchmark) is a comprehensive, verifiable office application benchmark. OPBM allows the an end-user as well as an experienced analyst to produce verifiable, real-world performance workloads utilizing popular office productivity software.

OPBM introduces new terminology to bundle benchmarking concepts so that office productivity workloads can be evaluated for real-world performance. A workload can be defined as an atom, molecule, scenario or suit.

OPBM provides a diversified tool set through 3 user interfaces

[**Simplified Run Interface**](#Simplified_Run_Interface) – an end-user interface which allows users to run preset benchmarks under closed, controlled conditions.

[**Developer Interface**](#Developer_Interface) – a developer interface to design, customize and capture office productivity tasks in such a way as to simulate “real-world” workloads for benchmark performance analysis.

[**Command Line Interface**](#Command_Line_Interface) - an end-user interface that allows the user to launch and execute OPBM scripts from a command prompt. The user can terminate the workload immediately from the command line or input files in a self-contained, controlled manner, utilizing the features of the GUI interface.

**Advantages of OPBM**

|  |  |
| --- | --- |
| **Customizable** | *OPBM’s unique design makes it customizable for the performance analyst or the hardware benchmarker.*  The end-user can choose which predefined workloads to include in his/her performance analysis, reorder the sequence of workloads in a run of the benchmark, test the workload sequence through a shorter, trial run before launching a full performance run. The developer can design and modify workloads, as well as customize his/her interface in OPBM, and capture errors/stuff through the command line interface. |
| **Flexible** | *OPBM’s unique design makes it extremely flexible for scripting purposes.*  The end-user and the developer can reorder workload sequences and determine which workloads to include in a run. The benchmark can focus on specific software, such as, internet browser, office products,…….l |
| **Terminology** | *OPBM’s unique design introduces new terminology to better identify benchmark workloads.*  **Worklet** **–** An operation or group of operations. It is the smallest unit of work that will generate a benchmark score.  **Atom** – The most basic bundle of worklets. Like atoms in nature, they may or may not be able to function independently. Worklets combine to form atoms.  **Molecule** – A group of atoms, such as Open Word atom, Load Word document atom and Close Word atom. A molecule can run independently.  **Scenario** – A group of molecules. A scenario may target performance analysis on a specific office productivity activity such as word processing. A scenario can run independently.  **Suite** – A group of scenarios. A suite can be used to define a complete benchmark for office productivity. A suite can run independently.  **HUD (Heads-Up-Display)** – Window active during a launch of OPBM. It displays information to show you which atom is running. |

|  |  |  |
| --- | --- | --- |
| **Scoring Methodology** | *OPBM’s scoring methodology makes it easy for end-users and experts alike to understand performance numbers.* | |
|  | **Official Run**  **Score** | An Official Run score (link to screen shot and definition???) is the arithmetic mean (average) of three OPBM scenarios. The Official Run score is the **average** of the three OPBM runs. The **Result Viewer** also provides the averages for all of the constituent operations and aggregations. |
|  | **How a score**  **is generated** | Each [worklet](#Worklet_definition) generates a score. This score is produced by dividing the time it took to complete the operation on the reference system (**tref**) by the time it took to complete the operation on the SUT (**tsut**) during the current run*.*  **tref\* / tsut**  **\****The current reference system is the Core i5 SUT.*  This ratio is then multiplied by 100 to produce a more human friendly number.  **score = 100 \* (tref / tsut )**  In other words, **the score is the SUT's percentage performance** of the reference system.  A score of  **100** means equal performance  **50** means half the performance  **200** means twice the performance of the reference system |
|  | **Aggregation**  **level score** | Scores are aggregated at each level by calculating the geometric mean of the component scores. For example,  ScoremyMolecule = GeometricMean( Scoreatom1, Scoreatom2,  Scoreatom3,… Scoreatomn)  The aggregation level scores are  **Atom** - The score is the geometric mean of internal operations.  **Molecule** -The score is the geometric mean of internal atoms.  **Scenario** - The score is the geometric mean of the internal molecules.  **Optional** **Suite** - The suite score will be the geometric mean of the internal scenario. A Suite allows for aggregation flexibility in scoring. |

**Minimum requirements to run OPBM**

* Oracle HotSpot Java Version 1.7.0 or newer
* Virtual Machine installed (JDK or JRE version 1.7.0 32-bit or 64-bit, preferred 64-bit)
* Internet Explorer 9 (will not work with IE8 or older)
* Microsoft Office 2010
* 24-bit video resolution of at least 1280x1024
* Windows 7

**How to set-up and install OPBM**

To install OPBM you will

1. [Create a git hub account](#Create_github_account)
2. Download the OPBM software from git hub to a local repository
3. [Disable the Windows 7 Login Screen.](#Disable_login_screen)
4. [Disable User Account Control](#Disable_user_account_control)
5. Run the opbm.jar file to start OPBM.

OPBM will install

* 7-zip
* Adobe Acrobat Reader X (version 10)
* Chrome
* Firefox
* Opera
* Safari

OPBM creates the following directories and documents when you install it.

Opbm

Java

Vs2010

OPBM\_Documentation.docx

readme.txt

The following default folders are created **in c:\users\user\documents\opbm** when OPBM is started.

|  |  |  |
| --- | --- | --- |
| **Folder created** | **What is in the folder** | **Sample file** |
| \results\csv\ | Benchmark results saved in csv format. |  |
| \results\xml | Benchmark results saved in xml format. Default file format for results. |  |
| \temp\ | Debug information and the opbm.dat files | debugInfo\_Oct\_04\_2011\_at17\_13\_44.xml  opbm.dat |
| \running\ | Manifest.xml  *When you launch the benchmark, OPBM generates a manifest.xml file. It records and accumulates everything necessary to complete the run, along with everything generated while executing scripts. This data exists in a single XML file called* ***manifest.xml****.* | manifest.xml |
| \settings\ |  |  |
| \scriptOutput\ | csv output file created by AutoIt for debug purposes | ieGoogleV8Times.csv |
| \scriptOutput\temp\ |  |  |

**GitHub documentation**

(See Git Hub Tutorial)

**Disable User Account Control**

1. Go to **Control Panel**.
2. Click on **User Accounts and Family Safety**.
3. Click on **User Accounts**. Make changes to your user account appears.
4. Click on **Change User Account Control settings**. The **User Account Control Settings** window appears.
5. Move the slider to **Never notify**. Click **OK**.
6. The **Action Center** window appears. You must restart the computer for the change to take place.

**Disable the Windows 7 login screen**

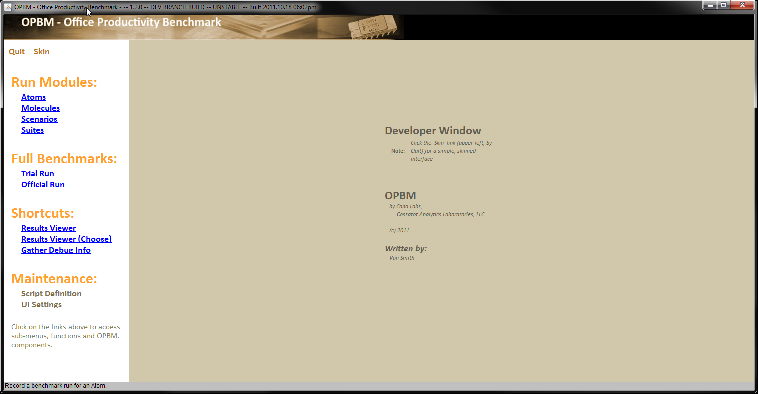
You must set windows to automatically log you in. This allows OPBM to reboot the system for an official benchmark run. The benchmark cannot capture appropriate scores if the system does not allow for automatic reboots.

1. Log into Windows 7 using an administrator account.
2. Click on the **Start Menu**. Type "netplwiz" in the search field. Click on "netplwiz" in the list. The **User Accounts** window appears.
3. Click on the user name to be logged in automatically.
4. Uncheck □ **Users must enter a user name and password to use this computer**.
5. Click on **Apply**. The **Automatically Log On** window appears. Enter your password in both **Password:** fields. Click **OK** to accept changes.

**Starting OPBM**

Double-click on **c:\opbm\java\opbm\opbm.jar** to start OPBM.

The **OPBM – Office Productivity Benchmark** main window appears.



**Parts of the OPBM – Office Productivity Benchmark Main Window**

**Quit** – Closes OPBM

**Skin** – Opens the **Simplified Run Interface**

**Run Modules:** *(These functions are part of the* [*Developer Interface*](#Developer_Interface)*.)*

Atoms – Opens the **Run Atoms:** window to test and customize atoms.

Molecules – Opens the **Run Molecules:** window to test and customize molecules.

Scenarios – Opens the **Run Scenarios:** window to test and customize scenarios.

Suites – Opens the **Run Suites:** window to test and customize suites.

Results Viewer – Opens the **OPBM – Results Viewer** window to view benchmark results.

**Full Benchmarks:** *(These functions are part of the* [*Simplified Run Interface*](#Simplified_Run_Interface)*.)*

Trial Run – Opens the **Trial Run** window to launch 1 full run of the OPBM benchmark.

Official Run – Opens the **Official Run** window to launch 3 full runs of the OPBM benchmark.

**Shortcuts:**

Results Viewer

Results Viewer (Choose)

Gather Debug Info

**Maintenance:**

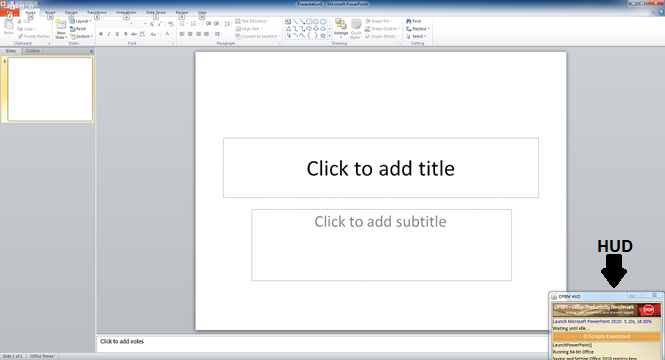
Script Definition – Opens the **Editing: and Maintenance:** panel. You can develop and edit atoms, molecules, scenarios and suites within OPBM.

UI Settings – Opens the **User interface settings** panel. You can change the settings here.

**The HUD (Heads Up Display)**

The HUD lists the execution order for a benchmark run.

The HUD appears when a run is launched. The HUD appears in the lower right screen of the window.



To terminate the run before completion, click on **STOP**.



When the benchmark run completes, the **OPBM HUD** window disappears and the **OPBM – Results Viewer** window appears with the results of the benchmark run.

**The Results Viewer**

[About the **OPBM – Results Viewer Window**](#Results_viewer)

[Parts of **the OPBM – Results Viewer Window**](#Results_viewer)

[Using **Show by: Times** button](#Using_the_Show_by_times_button)

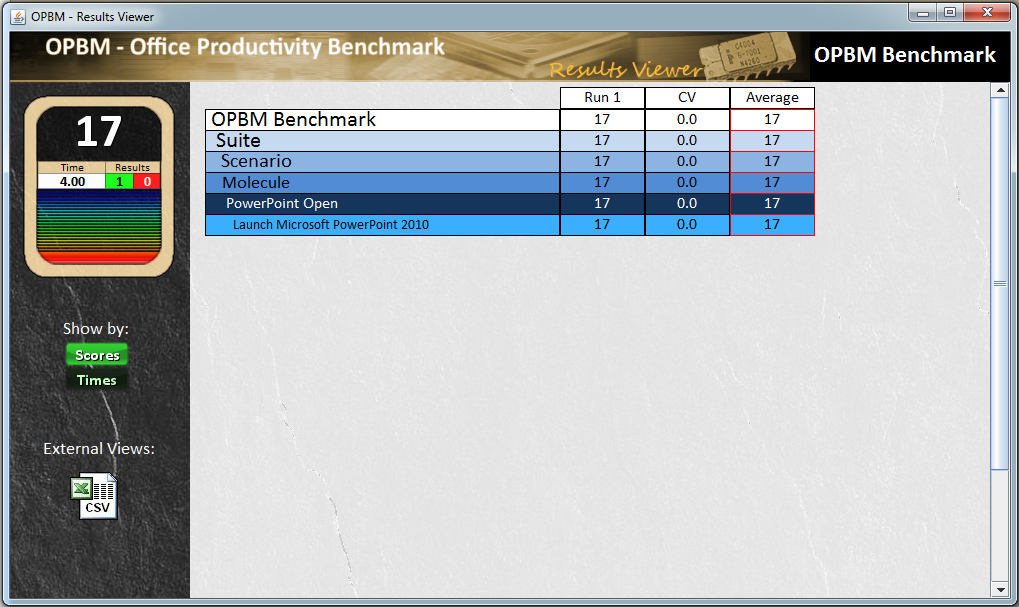
[Using **External Views: CSV** icon](#Using_the_external_views_csv_icon)

**About OPBM – Results Viewer Window**

The **OPBM – Results Viewer** appears at the end of a launch of the benchmark

or

if you click on the **Shortcuts: Results Viewer** link. The results appear by score. Display by score is the default setting.



**Parts of the OPBM – Results Viewer Window**

The left panel displays the benchmark score. A score of 100 is the same performance as the reference system. Currently, this is the corei5.

**Show by:**

**Scores** - Displays the results by score.

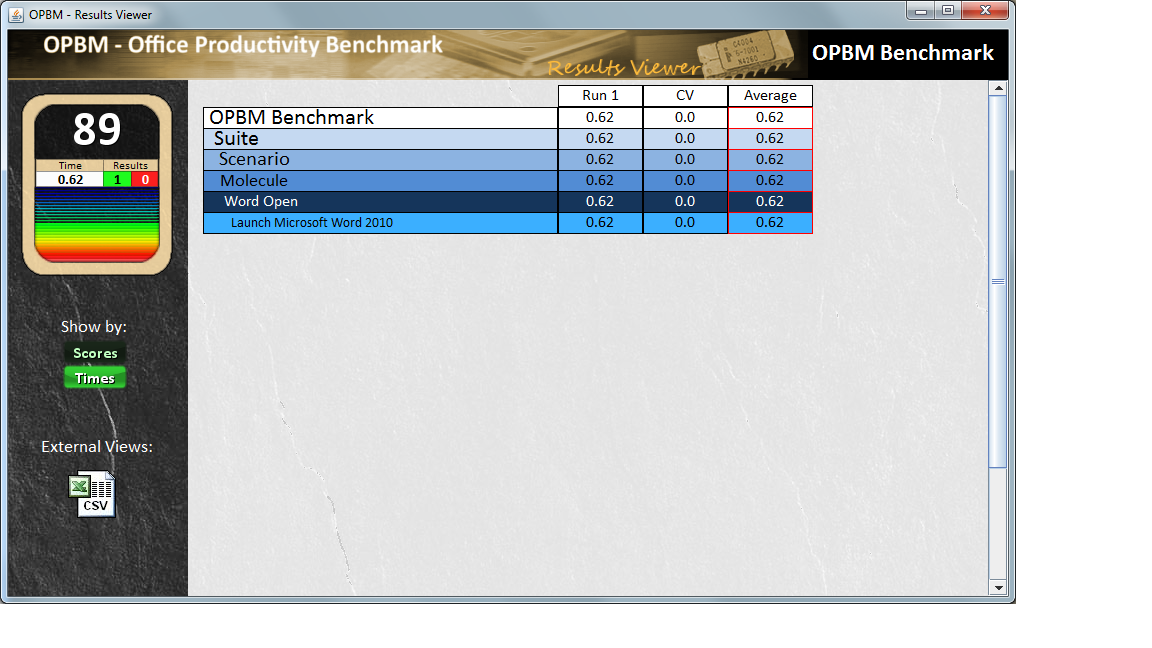
**Times -** Displays the results by time.

**External Views:**

CSV icon – Generates a CSV file in EXCEL.

**Using the Show by: Times button**

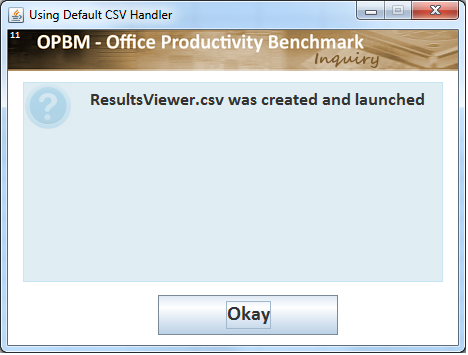
Click on **Show by: Times** to view the result times for the run.

****

**Using the External Views: icon**

Click on the icon to view the results in EXCEL. The file is a .cvs file. The **Using Default CSV Handler** window displays on top of the Excel spreadsheet. This window disappears after 15 seconds???? or Click the **OK** button.

??? new screen shot with EXCEL in background.



**The Simplified Run Interface**

[About the Simplified Run Interface](#About_the_simplified_run_interface)

[The Simplified Run Interface Main Window](#Simplified_run_interface_main_window)

[Using the **Trial Run** button](#Using_the_trial_run_button)

[Using the **Official Run** button](#Using_the_official_run_button)

[Using the **View Previous Results** button](#Using_the_view_previous_results_button)

[Using the **Developer Interface** button](#Using_the_developer_interface_button)

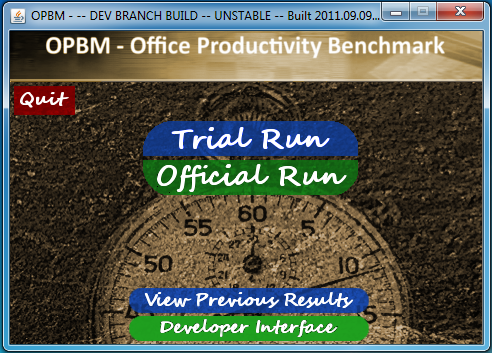
**About the Simplified Run Interface**

This will be the default interface upon release. It restricts the user to an official run or a trial run. An official run is 3 full runs of OPBM preceded by reboots before each run. A trial run is 1 full OPBM run with no reboots.

**The Simplified Run Interface Main Window**

Click on **Skin** from the **OPBM- Office Productivity Benchmark** main window.

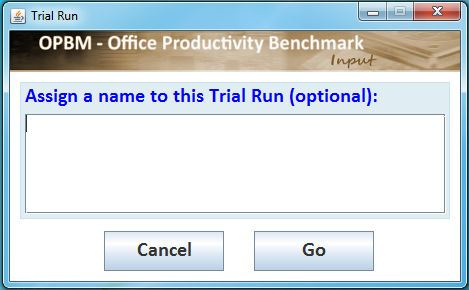
The **OPBM Simplified Run Interface** window appears.



|  |  |
| --- | --- |
| **Button** | **Function** |
| **Quit** | Quits the application. |
| **Trial Run** | Runs the benchmark one time. |
| **Official Run** | Runs the benchmark three times with reboots. |
| **View Previous Results** | Displays a file window to select previous results saved as an \*.xml file. The results can be viewed in the Results Viewer. ??? |
| **Developer Interface** | Returns user to the **OPBM – Office Productivity Benchmark** main window. |

**Using the Trial Run button**

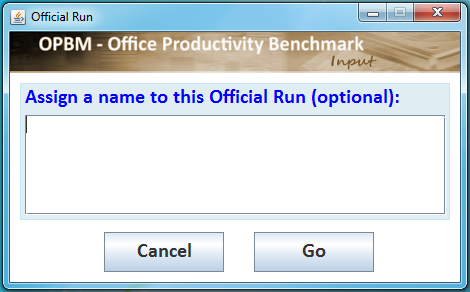
Click on the **Trial Run** button. The **Trial Run** window appears.



|  |  |
| --- | --- |
| **Saving Results** | To a user-named file…  Type the name of the file xxxx.xml. Click the **Go** button to start the run. The results are saved to the user-named .xml file in the \opbm\results\xml\ folder. |
|  | To the default results file…  Click on the **Go** button to start the run. The results are saved to results.xml in the \opbm\results\xml\ folder.. |
| **Returning to the Simplified Run Interface** | Click on the **Cancel** button. |

**Using the Official Run button**

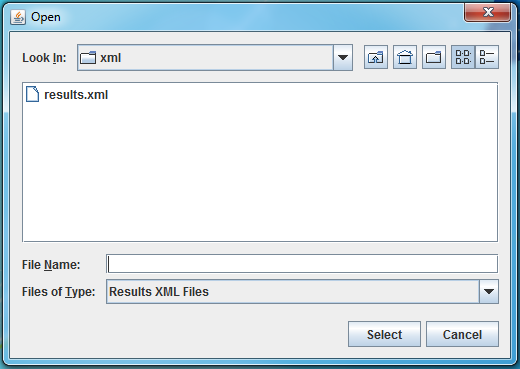
Click on the **Official Run** button. The **Official Run** window appears.



|  |  |
| --- | --- |
| **Saving Results** | To a user-named file…  Type the name of the file xxxx.xml. Click the **Go** button to start the run. The results are saved to the user-named .xml file in the \opbm\results\xml\ folder. |
|  | To the default results file…  Click on the **Go** button to start the run. The results are saved to results.xml in the \opbm\results\xml\ folder.. |
| **Returning to the Simplified Run Interface** | Click on the **Cancel** button. |

**Using the View Previous Results button**

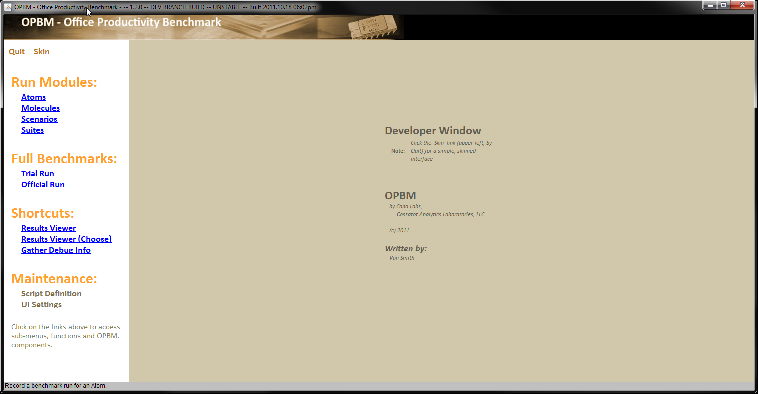
Click on the **View Previous Results** button. The **Open** window appears.



|  |  |
| --- | --- |
| **Viewing the results of a file** | Click on the file in the list, then click on the **Select** button. The **Results Viewer** window will appear.  Or  Double-click on the name of the file. The **Results Viewer** window will appear. |
| **Returning to the Simplified Run Interface** | Click on the **Cancel** button. |

**Using the Developer Interface button**

Click on the **Developer Interface** button to return to the **OPBM – Office Productivity Benchmark** main window.



**The Developer Interface**

[About the Developer Interface](#About_the_developer_interface)

[The Run Modules: Atoms](#Run_modules_atoms)

[The Run Modules: Molecules](#Run_modules_molecules)

[The Run Modules: Scenarios](#Run_modules_scenarios)

[The Run Modules: Suites](#Run_modules_suites)

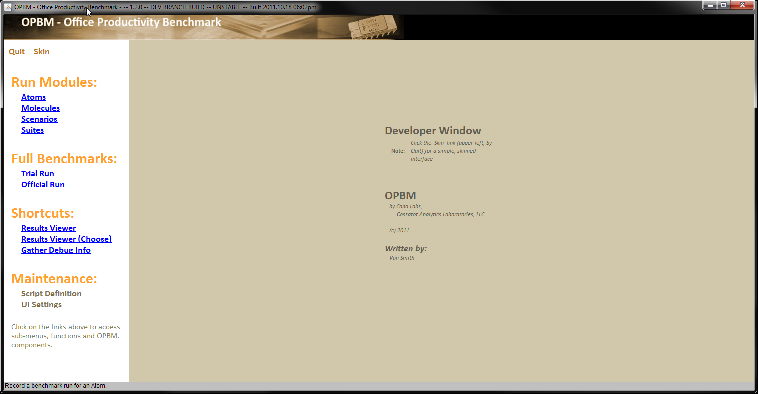
Maintenance: Script Definition

Maintenance: UI Settings

**About the Developer Run Interface**

The developer interface allows benchmark developers to customize, design and develop office productivity workloads for benchmarking. Workloads are defined by the number of tasks they include and can be bundled in several configurations to vary the sequencing and load usage.

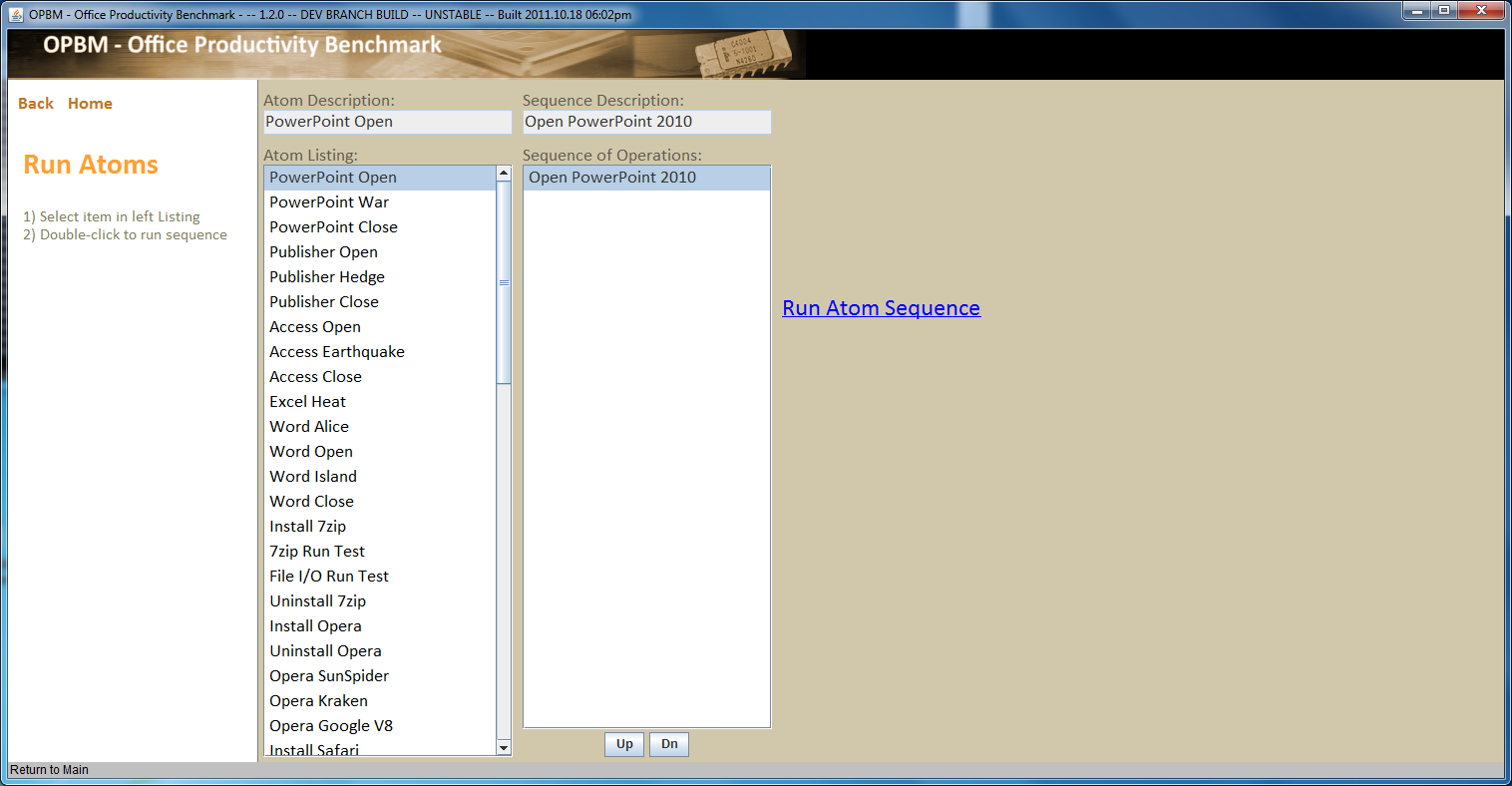
From the Developer Interface, you can run Atoms, Molecules, Scenarios, and Suites. The run produces a score.



**The Run Modules: Atoms window**

Click on **Atoms** under the **Run Modules** heading. The **Run Atoms** window appears.

<new screen>

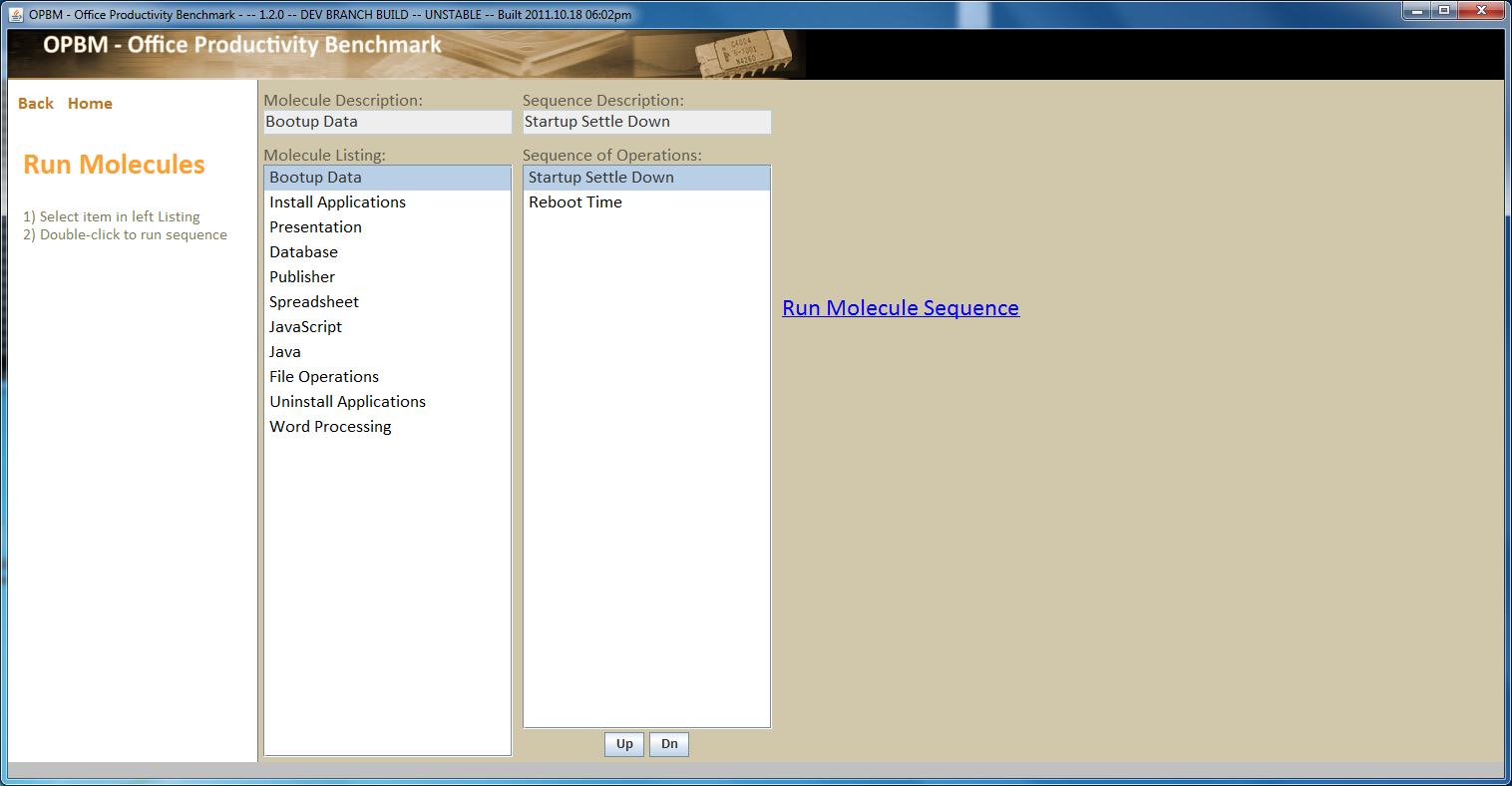


|  |  |
| --- | --- |
| **Running an atom** | Double-click on the atom name in the **Atom Listing:**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the screen.  OR |
|  | Click on the atom in the **Atom Listing:** to select it, then click on **Run Atom Sequence**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the display. |
| **Running an Atom Sequence with Iterations** | Click on the atom in the **Atom Listing:** to select it, then click on **Run Atom Sequence with iterations**. The Iterations window appears????. Type in the number of times that you want to run the atom. |

**The Run Modules: Molecules window**

Click on **Molecules** under the **Run Modules** heading. The **Run Molecules** window appears.

<new screen>

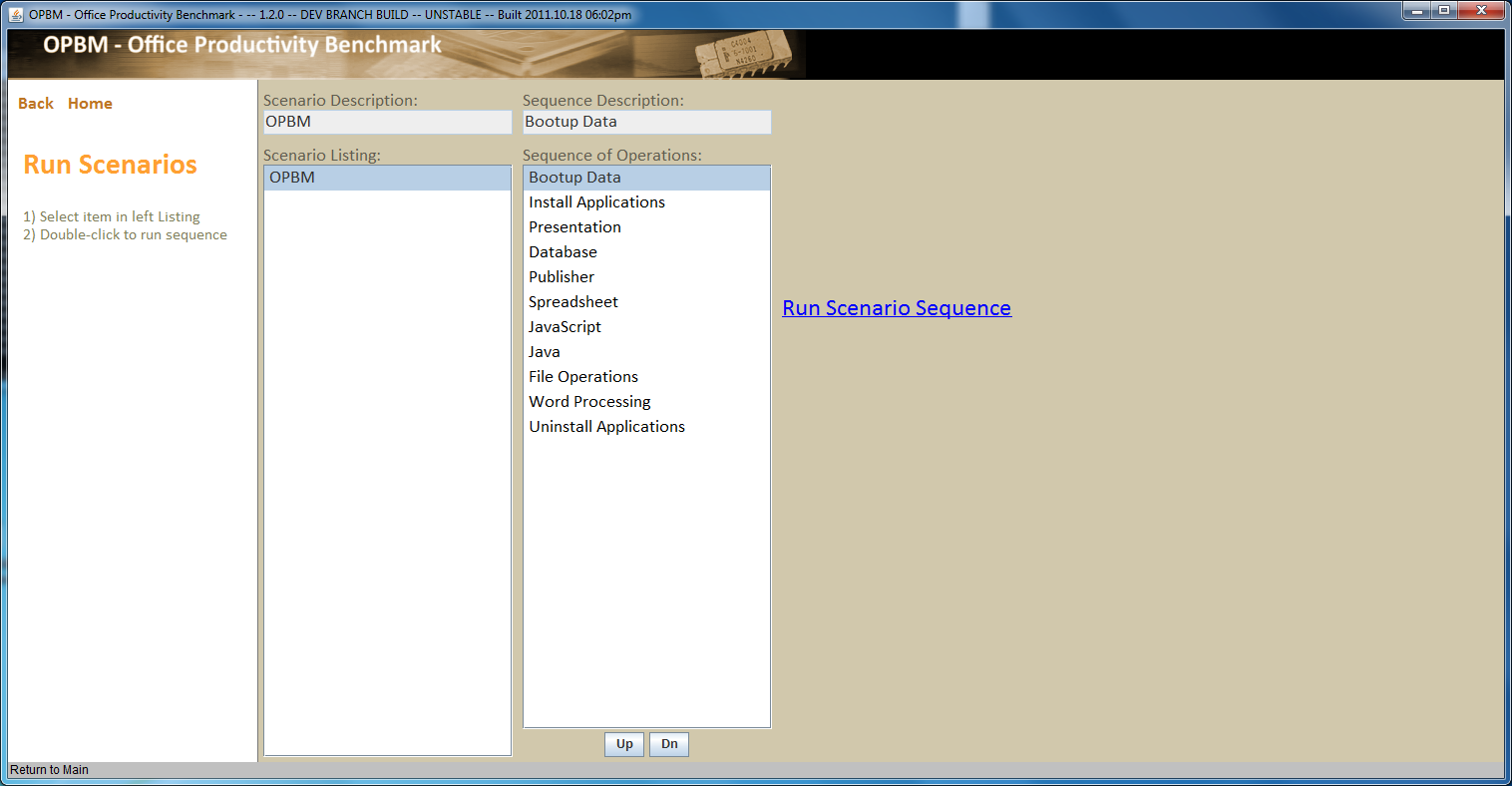


|  |  |
| --- | --- |
| **Running an molecule** | Double-click on the molecule name in the **Molecule Listing:**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the screen.  OR |
|  | Click on the atom in the **Molecule Listing:** to select it, then click on **Run Molecule Sequence**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the display. |
| **Running a molecule Sequence with Iterations** | Click on the atom in the **Atom Listing:** to select it, then click on **Run Atom Sequence with iterations**. The Iterations window appears????. Type in the number of times that you want to run the atom. |

**The Run Modules: Scenarios window**

Click on **Scenarios** under the **Run Modules** heading. The **Run Scenarios** window appears.

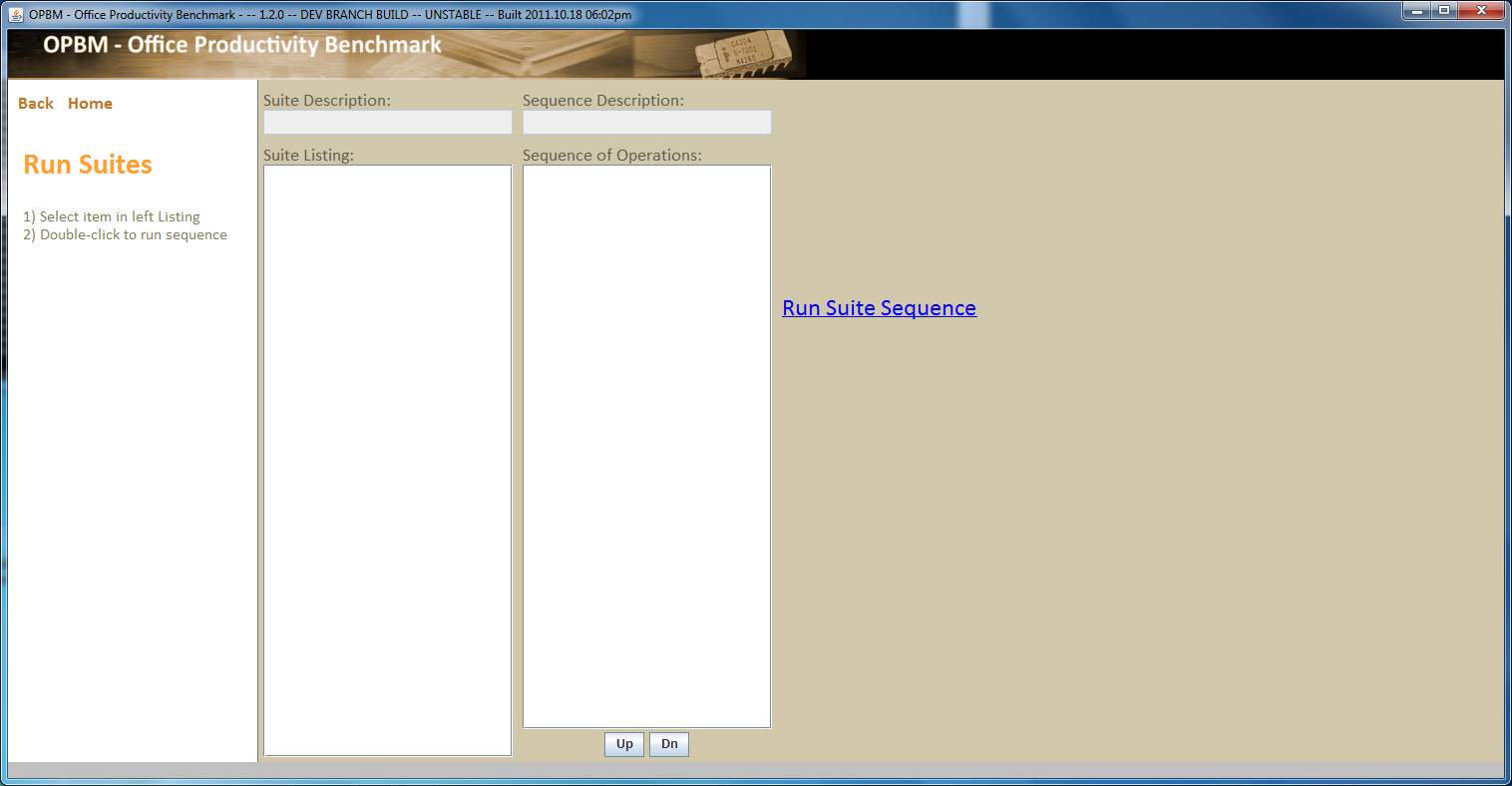
<new screen >



|  |  |
| --- | --- |
| **Running an scenario** | Double-click on the scenario name in the **Scenario Listing:**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the screen.  OR |
|  | Click on the atom in the **Scenario Listing:** to select it, then click on **Run Scenario Sequence**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the display. |

**The Run Modules: Suites window**

Click on **Suites** under the **Run Modules** heading. The **Run Suites** window appears.



|  |  |
| --- | --- |
| **Running an suite** | Double-click on the suite name in the **Suite Listing:**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the screen.  OR |
|  | Click on the atom in the **Suite Listing:** to select it, then click on **Run Suite Sequence**. OPBM launches the run. The HUD (Head-Up-Display) appears in the lower right corner of the display. |

**Maintenance: Script Definition**

Maintenance:Script Definition panel

[Editing: Atoms](#Editing_atoms)

[Editing: Molecules](#Editing_molecules)

[Editing: Scenarios](#Editing_scenarios)

Editing: Suites

Maintenance: View or Edit raw XML files

View Raw Files: View Panels.xml

View Raw Files: View Edits.sml

View Raw Files: ViewScripts.xml

Switch to Edit Mode

Click on Maintenance: Script Definition. The Editing: and Maintenance: panel appears.

Maintenance: Script Definition

Panels.xml - Defines menu navigation options

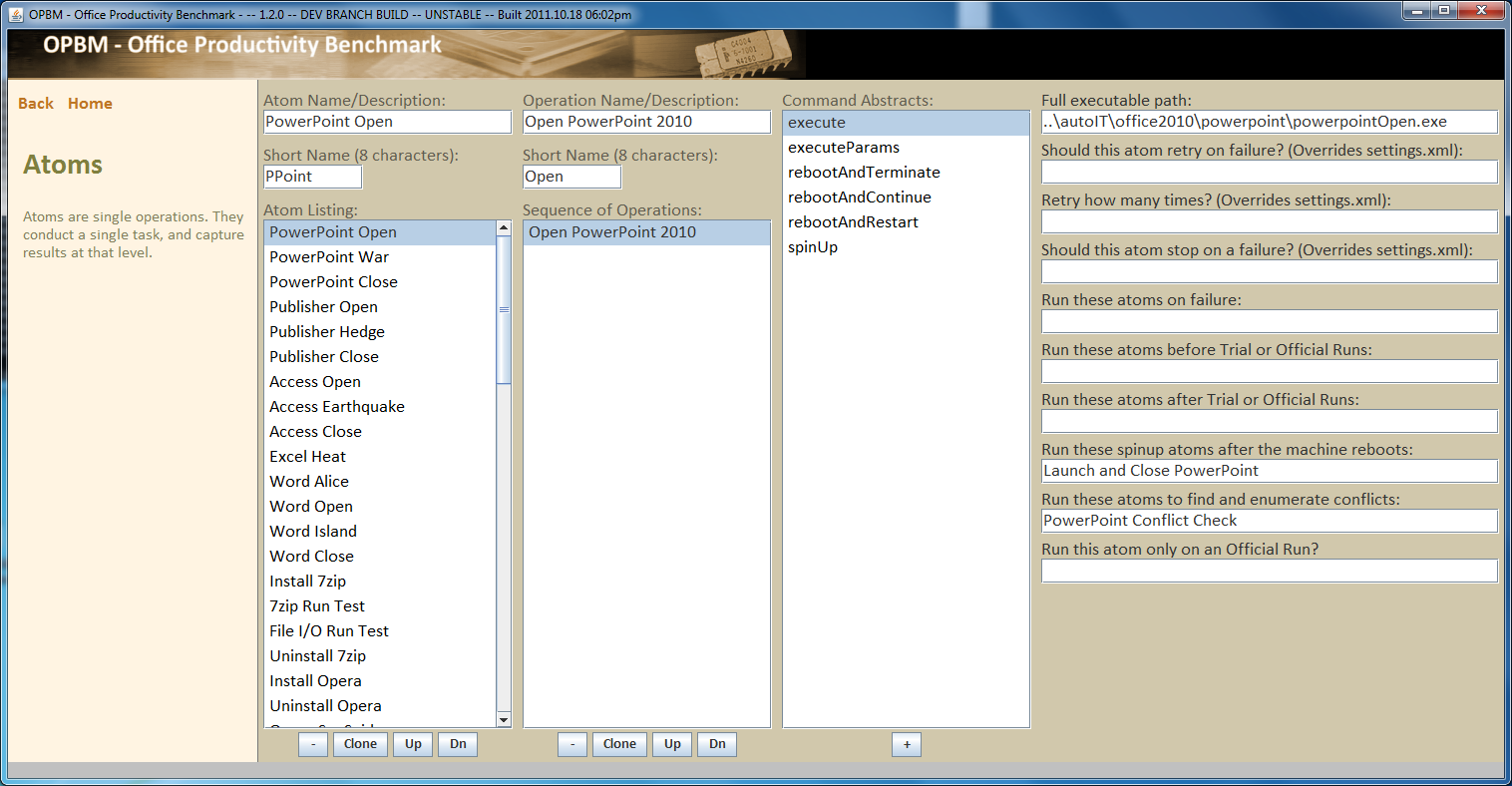
Edits.xml - Defines how OPBM edits data on-screen

Scripts.xml - Defines the scripts OPBM will execute

Maintenance: UI Settings

**Editing: Atoms**

Click on **Editing: Atoms**, the **Atoms** window appears.



Define buttons.

Describe list boxes

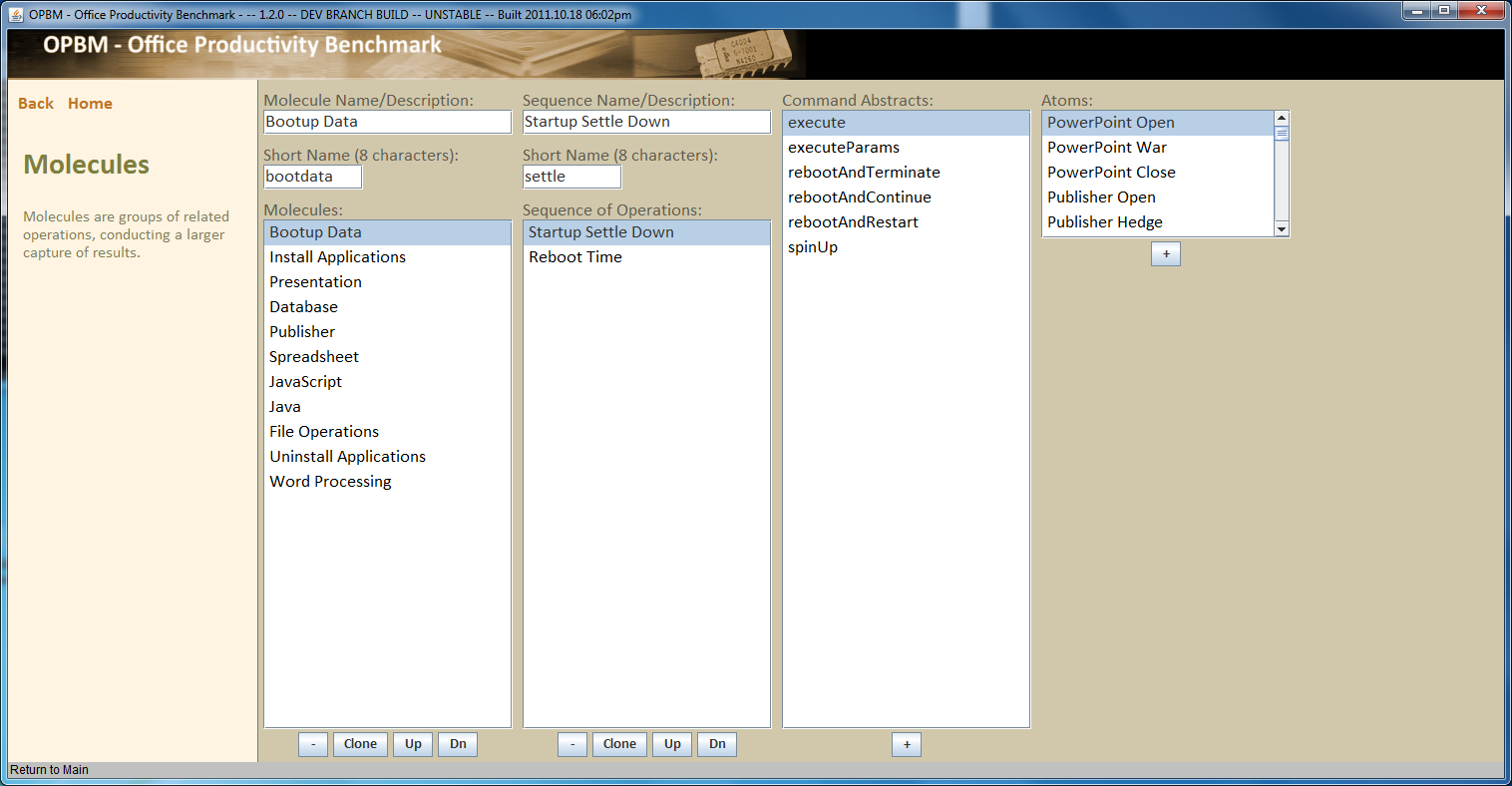
Describe Command Abstracts

**Command Abstracts** – in-between programming layer

Describe stuff to right…..

**Editing: Molecules**

Click on **Editing: Molecules**, the **Molecules** window appears.



Define buttons.

Describe list boxes

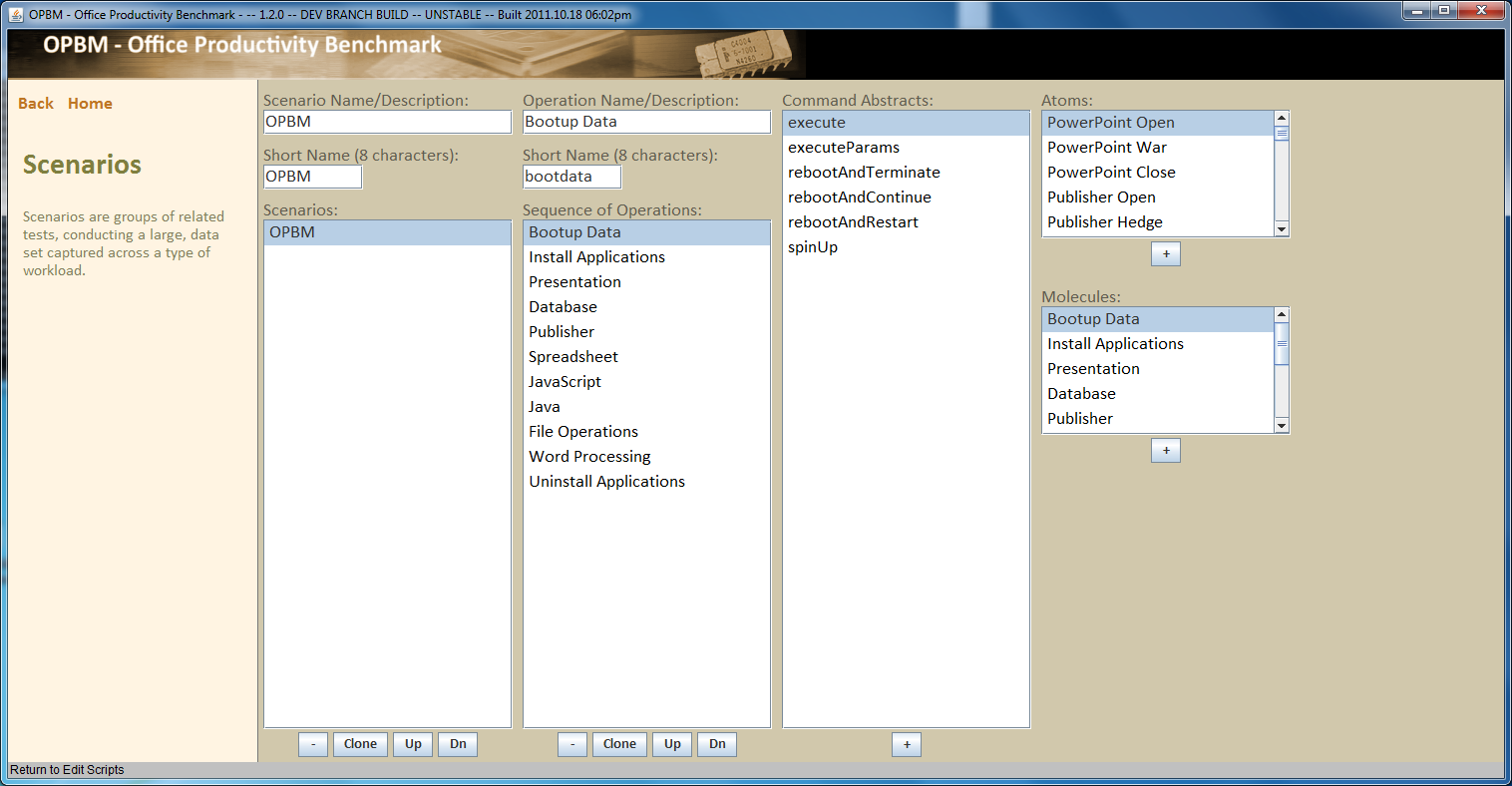
Describe Command Abstracts

**Command Abstracts** – in-between programming layer

Describe stuff to right…..

**Editing: Scenarios**

Click on **Editing: Scenarios**, the **Scenarios** window appears.



Define buttons.

Describe list boxes

Describe Command Abstracts

**Command Abstracts** – in-between programming layer

Describe stuff to right…..

**Editing: Suites**

Click on **Editing: Suites**, the **Suites** window appears.

GET SCREEN SHOT

Define buttons.

Describe list boxes

Describe Command Abstracts

**Command Abstracts** – in-between programming layer

Describe stuff to right…..

**Maintenance: View of Edit raw XML files**

Click on **Maintenance: View or Edit raw XML files**, the **View Raw Files:** panel appears

Click on View Panels,xml to display the Panels,xml file. The file appears. To close the file, click close.

Click on View Edits.xml to display the Edits.xml file.

Click on View Scripts.xml to display the Scripts.xml file.

Click on Switch to Edit Mode to edit the .xml file

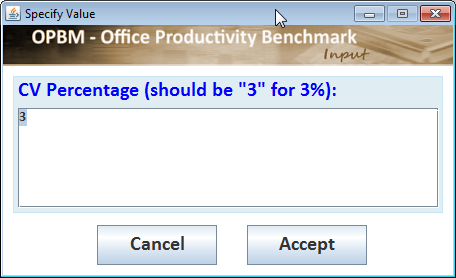
**Maintenance: UI Settings**

Click on **Maintenance: UI Settings**, the User Interface settings appear.

<new screen>

|  |  |  |  |
| --- | --- | --- | --- |
| **User Interface Option** | **Setting Type** | **What it does** | **Value for Setting** |
| **HUD settings** | Debug Info? | Displays on the bottom 4 lines of the HUD | YES – Displays Debug information  NO – Hides the Debug Information which makes the HUD smaller on the screen. |
|  | Show HUD? | Transparency of the HUD window | 25%  50%  75%  100%  NO –Does no display the HUD; sets Debug Info? To NO |
| **Benchmarks** | Retry Attempts: | Number of retry attempts before a failure is reported | 0-10 |
|  | Halt on Errors? | Stop the benchnmark because of error | YES – Stops benchmark on error  NO – Does not stop benchmark on error |
|  | Run Spinups? | Run spinup code after each reboot  (What is spinup code) | YES – Run spinup code  NO – Do not run spinup code |
|  | Uninstall on failure? | Failure of a run executes the uninstall atoms | YES – Run uninstall atoms to uninstall applications if the run fails.  NO – Do not run the uninstall atoms if the run fails. |
| **Results Viewer** | Results Viewer CV: | Percent threshold for highlighted display in the **Results Viewer** CV column | Type in value in the **Specify Value** window.\* |

\***Specify Value** window



**The Command Line Interface**

[About The Command Line Interface](#About_the_command_line_interface)

[Command Line Syntax](#Command_line_syntax)

[Running OPBM from the command line](#Running_OPBM_from_the_command_line)

[Executing scripts from the command line](#Executing_scripts_from_the_command_line)

[Benchmarking across multiple machines](#Benchmarking_across_multiple_machines)

**About The Command Line Interface**

The command line interfaces lets benchmark developers run OPBM from the command line. This flexibility allows the developer to …..

**Command Line syntax**

[Running OPBM from the command line](#Running_OPBM_from_the_command_line)

[Executing scripts from the command line](#Executing_scripts_from_the_command_line)

[Benchmarking across multiple machines](#Benchmarking_across_multiple_machines)

Default directories that OPBM uses when it is installed.

Click [here](#Default_directories_for_OPBM) to see the default directories.

**Running OPBM from the command line**

| **ACTION** | **Command** | **Good to know** |
| --- | --- | --- |
| **Run OPBM** | cd c: \location\of\jar  or  c:\path\to\\java\exe – jar opbm.jar |  |
| **Create a shortcut** | C:\path\to\opbm.jar |  |
| **Execute a full Trial Run** | java –jar opbm.jar –trial |  |
| **Execute an Official Run** | java –jar opbm.jar –official |  |
| **Run the Simplified Run Interface** | java –jar opbm.jar –simple  or  java –jar opbm.jar –skin |  |
| **Run the Developer Interface** | java –jar opbm.jar –developer |  |
| **Execute OPBM without the program terminating on completion of a script** | java –jar opbm.jar –noexit |  |
| **Assign a name to the results file (?) for the run** | java –jar opbm.jar –name:”*Give the run a name*” |  |
| **Override the default Oracle system property for java.home** | java –jar opbm.jar –home:”c:\full\path\to\jave.exe” |  |
| **Automatically force an OPBM restart** | java –jar opbm.jar -restart | *The option is used internally as part of the reboot-and-continue operations of an Official Run three-pass benchmark. When present OPBM will attempt to continue processing the manifext.xml file in the c:\users\user\opbm\running directory from where it left off at the reboot. To prevent OPBM from restarting a run automatically, clear out all files in the c:\users\user\opbm\running directory* |

**Executing scripts from the command line**

|  |  |  |
| --- | --- | --- |
| **Action** | **Command** | **Good to know!** |
| **Execute a specific atom script** | **java –jar opbm.jar –atom:***script\_name*  Ex. **java –jar opbm.jar –atom:**wordalice | *Strip the spaces from the script name before adding it to the command line.*  *Ex:* **Word Alice** *becomes* **wordalice** |
| **Execute multiple atoms** | **java –jar opbm.jar –atom:***script\_name*  **–atom:***script\_name*  –**atom:***script\_name* | *There is no limit on the number of atoms you can run* |
| **Execute iterations for a atom script** | **java –jar opbm.jar -atom(***iteration\_count***):***script\_name*  Ex: **java –jar opbm.jar – atom(10):**wordalice | *You can execute multiple atoms with multiple iterations.* |
| **Execute molecule, suite, scenario** | Syntax similar to atom  **java –jar opbm.jar**  **-molecule(***iteration\_count***):***script\_name*  **-scenario(***iteration\_count***):***script\_name*  **-suite(***iteration\_count***):***script\_name* | NOT YET AVAILABLE |
| **Launch the JAR without specifying the java executable** | **opbm.jar –atom:**wordalice  or  **opbm.jar -atom:**wordalice **-atom:**anotheratom  or  **opbm.jar -atom(10):**wordalice **-atom(10):**anotheratom |  |

**Benchmarking across multiple machines**

|  |  |
| --- | --- |
| **What you need to do** | To execute similar benchmarks on multiple machines, create a file which contains the sequence of benchmarks to run in a specified order. . OPBM uses this file at startup to execute identical benchmarks across a host of machines.  1. Create the INPUT FILE  2. Copy the file to the SUTS or network location  3. Access the file through script or network |
| **INPUT FILE syntax** | Each entry must be stored on a separate line in the file. Use the same syntax as the command line syntax, -**atom:***script\_name*  Sample input file  Sample.txt  -atom:wordalice  -atom(100):gpuheavy  -atom(100):opencltests  -atom(100):publisherintensive  -atom(500):wordintensive  -atom(200):excelintensive |
| **Running the INPUT FILE** | java -jar opbm.jar @c:\path\to\sample.txt  OPBM interprets the input from the sample.txt file as though the user repeatedly typed entries on the command line, one after the other. The line entries are added in sequence as they appear on the command line.  Any combination of command-line options and command-line input files can be specified, allowing for a concatenation of both directly specified entries on the command line, as well as input from a file. There are no limits to how many entries can be specified in this way, and the entries will be queued in the order stated on the command line. |
| **Examples** | java -jar opbm.jar -atom:wordalice @sample.txt  java -jar opbm.jar @sample.txt -atom(100):wordalice  java -jar opbm.jar -atom:wordalice @sample.txt -atom(100):wordalice  OR  opbm.jar -atom:wordalice @sample.txt  opbm.jar @sample.txt -atom(100):wordalice  opbm.jar -atom:wordalice @sample.txt -atom(100):wordalice |

**OPBM Harness Configuration**

OPBM Files

|  |  |
| --- | --- |
| **.xml files which drive OPBM** | edits.xmls- defines the way OPBM edits data on-screen  panels.xml- defines menu navigation options  scripts.xml- defines the scripts OPBM will execute  ! These files must be setup correctly or OPBM will not launch properly. OPBM will enter an error-correcting edit mode in which the raw XML files are loaded for editing when not setup correctly. |
|  |  |

Output Files and Processing….

**Output Files and Processing**

As of the first June 27, 2011 release, continued through the August 22, 2011 release, OPBM internally recognizes four types of line items written to stdout or stderr. These are reported by the scripts executing, and are used to update the heads-up-display, and to capture some timing data f

or debugging:

**timing** Conveys timing information to OPBM. Must have the form “Workload description,timing,percentage”. Example: “Launch Microsoft Word,1.5733983892,89.8329821602”. Appears in blue.

**debug** Conveys debug information (shows up in the debug portion of the heads-up-display, which are the bottom 4 lines).

**status** Conveys status information (shows up in the status portion of the heads-up-display, which are the two lines above the middle gray portion).

**error** Conveys error information, which is usually terminal. In future versions, if the keyword “terminate” is found on the error string, OPBM will automatically terminate the current benchmark test and continue with any more that are scheduled. Appears in red.

**overhead** Conveys timing information related to overhead processing, such as the time required to launch an application, or type in keystrokes to access a file path or URL.

**filter** Conveys filter tags for the executing atom. These are used in the post-processor and Results Viewer to aggregate scores in manifest.xml’s <aggregate> section in a <byFilter> division (planned future feature as of 08/31/2011).

**tags** Conveys worklet tags, which are associated only with the immediately-following timing event. These tags allow individual worklets to be aggregated into similar scores in the manifest.xml’s <aggregate> section in a <byTags> division (planned future feature as of 08/31/2011).

When a benchmark run is requested, OPBM generates a manifest.xml file. It records and accumulates everything necessary to complete the run, along with everything generated while executing scripts. This data exists in a single XML file called **manifest.xml**. This file is written out to a relative location based on the current user, such as:

**C:\Users\username\Documents\opbm\running\manifest.xml**

OPBM also auto-generates two output files following the successful completion of a benchmark, which are **results.xml** and **results.csv**, which are located:

**C:\Users\username\Documents\opbm\results\xml\results.xml**

**C:\Users\username\Documents\opbm\results\csv\results.csv**

The scripts themselves may also write content directly to a designated output directory, though this is not required. The script’s output directory is located:

**C:\Users\username\Documents\opbm\scriptOutput\**

OPBM records each timing line as it was generated. If multiple instances of the same test were run, then OPBM automatically computes the min, max, average, geometric mean, and coefficient of variation in the results. OPBM also automatically appends a “Total” line to the reported timing lines as well in the output CSV. Only *timing* timing events are included in the output, and not the *overhead* lines which also include times.

**Note:** Iteration requests present themselves internally as though the single atom were run successively for the iteration count, meaning each execution by iteration produces its own separate set of summation values, which are recorded within the manifest.xml file. OPBM will automatically sum and average these values into the <aggregate> section entitled <byAtom>. These scores are also conveyed into the results.xml file for viewing in the Results Viewer, with a single line included for every iteration.

**Problems running OPBM**

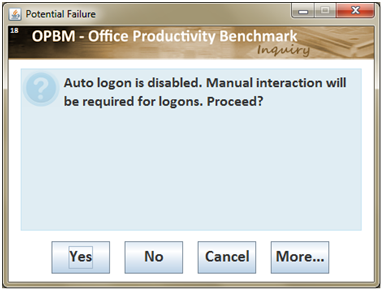
If you cannot launch an official run, then …..

[Auto-Logon Disable](#Auto_logon_disable)

[Conflicts and Resolutions Window](#Conflicts_and_resolutions)

**Auto Logon Disable**

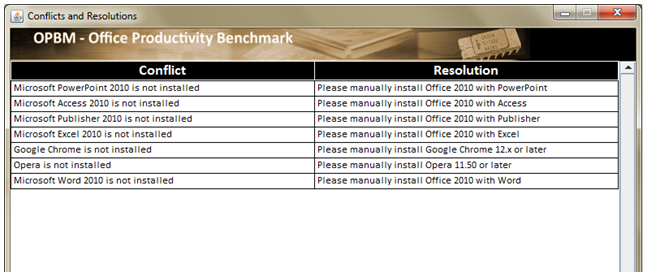
If you do not [disable the Windows 7 login screen](#Disable_windows_login), this window appears when you attempt an official run.



|  |  |
| --- | --- |
| **Button** | **Function** |
| **Yes** | **!** Launches the official run. An official run is 3 iterations of the benchmark. It requires the system to reboot. You will be required to login to Windows when a reboot occurs in order for the benchmark to continue. This will result in skewed scoring times. |
| **No** | Run will not launch. |
| **Cancel** | Cancels the launch. |
| **More…** | Displays the directions to disable the Window 7 login screen. |

**Conflicts and Resolutions**

If you launch an official run and the proper office applications are not installed ([see minimum requirements to run OPBM](#Minimum_requirements_to_run_OPBM)), the **Conflicts and** Resolutions window appears.



Manually install the application before attempting to launch an official run.

**Getting Help**

|  |  |  |  |
| --- | --- | --- | --- |
| Installation |  |  |  |
| GitHub | Van Smith | [van@canalabs.com](mailto:van@canalabs.com) | (479) 216-3461 |
| Bugs |  |  |  |
| Internal AMD set-up | Jim??? | [jim@amd.com](mailto:jim@amd.com) | (512) 123-1234 |
|  |  |  |  |
|  |  |  |  |

END NOTES TO BE DELETED, EVENTUALLY

-----------------------------------------------------------------

Definitions

**Flow Control** – directives to handle looping, conditional execution, script commands, etc Definitions Flow Control #13.PNG

Some buttons....

**Command Abstracts** – in-between programming layer - Definitions Command Abstracts #14.PNG

Some buttons.... what are the Up and down buttons for?

????WHAT ABOUT EXECUTABLE SCRIPT DIRECTORY LAYOUT FOR DEVELOPMENT from doc OPBM\_Executable\_Script\_Directory\_Layout.docx