
D-Wave homebase

Release 1.0.1

Jul 30, 2018

Contents:

1	homebase	1
2	License	3
3	Documentation	5
4	Indices and tables	15
	Python Module Index	17

CHAPTER 1

homebase

An API for getting platform independent paths to user data and configuration folders, with support for [virtualenv](#)

CHAPTER 2

License

Released under the Apache License 2.0 (see License).

3.1 hombase module

This module provides an API for determining application specific directories for data, config, logs, etc.

public module members:

<code>user_data_dir(app_name, app_author[, ...])</code>	Return the full path to the user data dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.
<code>user_config_dir(app_name, app_author[, ...])</code>	Return the full path to the user config dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.
<code>user_cache_dir(app_name, app_author[, ...])</code>	Return the full path to the user cache dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.
<code>user_state_dir(app_name, app_author[, ...])</code>	Return the full path to the user state dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.
<code>user_logs_dir(app_name, app_author[, ...])</code>	Return the full path to the user log dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.
<code>site_data_dir(app_name, app_author[, ...])</code>	Return the full path to the OS wide data dir for this application.
<code>site_config_dir(app_name, app_author[, ...])</code>	Return the full path to the OS wide config dir for this application.
<code>site_data_dir_list(app_name, app_author[, ...])</code>	Return the list of full path to the OS wide data directories for this application.

Continued on next page

Table 1 – continued from previous page

<code>site_config_dir_list(app_name, app_author[, ...])</code>	Return the list of full path to the OS wide data directories for this application.
--	--

This code is inspired by and builds on top of code from <http://github.com/ActiveState/appdirs>

class Platform

Bases: `enum.Enum`

WINDOWS = 1

MACOS = 2

POSIX = 3

user_data_dir(*app_name*, *app_author*, *version=None*, *roaming=False*, *use_virtualenv=True*, *create=True*)

Return the full path to the user data dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.

If using a virtualenv, the path returned is `/path/to/virtualenv/data/app_name`

Typical user data directories are:

- Mac OS X: `~/Library/Application Support/<app_name>`
- Unix: `~/.local/share/<app_name>` # or `$XDG_DATA_HOME/<app_name>`, if defined.
- Win XP (not roaming): `C:\Documents and Settings\<username>\Application Data\<app_author>\<app_name>`
- Win XP (roaming): `C:\Documents and Settings\<username>\Local Settings\Application Data\<app_author>\<app_name>`
- Win 7 (not roaming): `C:\Users\<username>\AppData\Local\<app_author>\<app_name>`
- Win 7 (roaming): `C:\Users\<username>\AppData\Roaming\<app_author>\<app_name>`

For Unix, we follow the XDG spec and support `$XDG_DATA_HOME`. That means, by default `~/.local/share/<AppName>`.

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user data path.
- **app_author** (*str*) – Only used in Windows when not in a virtualenv, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the `app_name`.
- **roaming** (*bool*) – roaming appdata directory. That means that for users on a Windows network setup for roaming profiles, this user data will be synchronized on login. See [http://technet.microsoft.com/en-us/library/cc766489\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc766489(WS.10).aspx) for a discussion of issues.
- **use_virtualenv** (*bool*) – If True and we're running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the user data dir for this application.

Return type `str`

user_config_dir (*app_name*, *app_author*, *version=None*, *roaming=False*, *use_virtualenv=True*, *create=True*)

Return the full path to the user config dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.

If using a virtualenv, the path returned is `/path/to/virtualenv/config/app_name`

Typical user config directories are:

- Mac OS X: same as `user_data_dir`
- Unix: `~/.config/<AppName>` # or in `$XDG_CONFIG_HOME`, if defined
- Win *: same as `user_data_dir`

For Unix, we follow the XDG spec and support `$XDG_CONFIG_HOME`. That means, by default `~/.config/<AppName>`.

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the `app_name`.
- **roaming** (*bool*) – roaming appdata directory. That means that for users on a Windows network setup for roaming profiles, this user data will be synchronized on login. See [http://technet.microsoft.com/en-us/library/cc766489\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc766489(WS.10).aspx) for a discussion of issues.
- **use_virtualenv** (*bool*) – If True and we're running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the user config dir for this application.

Return type `str`

user_cache_dir (*app_name*, *app_author*, *version=None*, *use_virtualenv=True*, *create=True*)

Return the full path to the user cache dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS's convention if it doesn't.

If using a virtualenv, the path returned is `/path/to/virtualenv/cache/app_name`

Typical user cache directories are:

- Mac OS X: `~/Library/Caches/<AppName>`
- Unix: `~/.cache/<AppName>` (XDG default)
- Win XP: `C:\Documents and Settings\<username>\Local Settings\Application Data\<AppAuthor>\<AppName>\Cache`
- Vista: `C:\Users\<username>\AppData\Local\<AppAuthor>\<AppName>\Cache`

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the `app_name`.

- **use_virtualenv** (*bool*) – If True and we’re running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the user cache dir for this application.

Return type *str*

user_state_dir (*app_name*, *app_author*, *version=None*, *roaming=False*, *use_virtualenv=True*, *create=True*)

Return the full path to the user state dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS’s convention if it doesn’t.

If using a virtualenv, the path returned is `/path/to/virtualenv/state/app_name`

Typical user state directories are:

- Mac OS X: same as `user_data_dir`
- Unix: `~/.local/state/<AppName>` # or in `$XDG_STATE_HOME`, if defined
- Win *: same as `user_data_dir`

For Unix, we follow this Debian proposal <https://wiki.debian.org/XDGBaseDirectorySpecification#state> to extend the XDG spec and support `$XDG_STATE_HOME`. That means, by default `~/.local/state/<AppName>`.

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the `app_name`.
- **roaming** (*bool*) – roaming appdata directory. That means that for users on a Windows network setup for roaming profiles, this user data will be synchronized on login. See [http://technet.microsoft.com/en-us/library/cc766489\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc766489(WS.10).aspx) for a discussion of issues.
- **use_virtualenv** (*bool*) – If True and we’re running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the user state dir for this application.

Return type *str*

user_logs_dir (*app_name*, *app_author*, *version=None*, *use_virtualenv=True*, *create=True*)

Return the full path to the user log dir for this application, using a virtualenv location as a base, if it exists, and falling back to the host OS’s convention if it doesn’t.

If using a virtualenv, the path returned is `:bash:/path/to/virtualenv/log/app_name“`

Typical user log directories are:

- Mac OS X: `~/Library/Logs/<AppName>`
- Unix: `~/.cache/<AppName>/log` # or under `$XDG_CACHE_HOME` if defined
- Win XP: `C:\Documents and Settings\<username>\Local Settings\Application Data\<AppAuthor>\<AppName>\Logs`
- Vista: `C:\Users\<username>\AppData\Local\<AppAuthor>\<AppName>\Logs`

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the app_name.
- **use_virtualenv** (*bool*) – If True and we’re running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the user log dir for this application.

Return type *str*

site_data_dir (*app_name*, *app_author*, *version=None*, *use_virtualenv=True*, *create=False*)

Return the full path to the OS wide data dir for this application.

Typical site data directories are:

- Mac OS X: /Library/Application Support/<AppName>
- Unix: /usr/local/share/<AppName> or /usr/share/<AppName>
- Win XP: C:\Documents and Settings\All Users\Application Data\<AppAuthor>\<AppName>
- Vista: (Fail! "C:\ProgramData" is a hidden *system* directory on Vista.)
- Win 7: C:\ProgramData\<AppAuthor>\<AppName> # Hidden, but writeable on Win 7.

For *nix, this is using the \$XDG_DATA_DIRS default.

Note: On linux, the \$XDG_DATA_DIRS environment variable may contain a list. *site_data_dir* returns the first element of this list. If you want access to the whole list, use *site_data_dir_list()*

<p>Warning: Do not use this on Windows Vista. See the Vista-Fail note above for why.</p>

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the app_name.
- **use_virtualenv** (*bool*) – If True and we’re running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the site data dir for this application.

Return type `str`

site_data_dir_list (*app_name*, *app_author*, *version=None*, *use_virtualenv=True*, *create=False*)

Return the list of full path to the OS wide data directories for this application.

Typical site data directories are:

- Mac OS X: /Library/Application Support/<AppName>
- Unix: /usr/local/share/<AppName> or /usr/share/<AppName>
- Win XP: C:\Documents and Settings\All Users\Application Data\<AppAuthor>\<AppName>
- Vista: (Fail! "C:\ProgramData" is a hidden *system* directory on Vista.)
- Win 7: C:\ProgramData\<AppAuthor>\<AppName> # Hidden, but writeable on Win 7.

For *nix, this is using the \$XDG_DATA_DIRS default.

Warning: Do not use this on Windows Vista. See the Vista-Fail note above for why.

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the app_name.
- **use_virtualenv** (*bool*) – If True and we're running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns A list to the full paths for site data directories for this application.

Return type `list`

site_config_dir (*app_name*, *app_author*, *version=None*, *use_virtualenv=True*, *create=False*)

Return the full path to the OS wide config dir for this application.

Typical site data directories are:

- Mac OS X: /Library/Application Support/<AppName>
- Unix: /usr/local/share/<AppName> or /usr/share/<AppName>
- Win XP: C:\Documents and Settings\All Users\Application Data\<AppAuthor>\<AppName>
- Vista: (Fail! "C:\ProgramData" is a hidden *system* directory on Vista.)
- Win 7: C:\ProgramData\<AppAuthor>\<AppName> # Hidden, but writeable on Win 7.

For *nix, this is using the \$XDG_DATA_DIRS default.

Note: On linux, the \$XDG_CONFIG_DIRS environment variable may contain a list. *site_config_dir* returns the first element of this list. If you want access to the whole list, use *site_config_dir_list()*

Warning: Do not use this on Windows Vista. See the Vista-Fail note above for why.

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the app_name.
- **use_virtualenv** (*bool*) – If True and we're running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns the full path to the site config dir for this application.

Return type *str*

site_config_dir_list (*app_name, app_author, version=None, use_virtualenv=True, create=False*)

Return the list of full path to the OS wide data directories for this application.

Typical site data directories are:

- Mac OS X: /Library/Application Support/<AppName>
- Unix: /usr/local/share/<AppName> or /usr/share/<AppName>
- Win XP: C:\Documents and Settings\All Users\Application Data\<AppAuthor>\<AppName>
- Vista: (Fail! "C:\ProgramData" is a hidden *system* directory on Vista.)
- Win 7: C:\ProgramData\<AppAuthor>\<AppName> # Hidden, but writeable on Win 7.

For *nix, this is using the \$XDG_DATA_DIRS default.

Warning: Do not use this on Windows Vista. See the Vista-Fail note above for why.

Parameters

- **app_name** (*str*) – Name of the application. Will be appended to the base user config path.
- **app_author** (*str*) – Only used in Windows, name of the application author.
- **version** (*str*) – If given, the application version identifier will be appended to the app_name.

- **use_virtualenv** (*bool*) – If True and we’re running inside of a virtualenv, return a path relative to that environment.
- **create** (*bool*) – If True, the folder is created if it does not exist before the path is returned.

Returns A list to the full paths for site data directories for this application.

Return type `list`

3.2 License

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

“License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

“Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

“Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, “control” means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

“You” (or “Your”) shall mean an individual or Legal Entity exercising permissions granted by this License.

“Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

“Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

“Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

“Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the

purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service

marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

This project includes code from the appdirs python package <https://github.com/ActiveState/appdirs> under the following license:

This is the MIT license

Copyright (c) 2010 ActiveState Software Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CHAPTER 4

Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)
- [Glossary](#)

h

`homebase.homebase`, [5](#)

H

homebase.homebase (module), [5](#)

M

MACOS (Platform attribute), [6](#)

P

Platform (class in homebase.homebase), [6](#)

POSIX (Platform attribute), [6](#)

S

site_config_dir() (in module homebase.homebase), [10](#)

site_config_dir_list() (in module homebase.homebase),
[11](#)

site_data_dir() (in module homebase.homebase), [9](#)

site_data_dir_list() (in module homebase.homebase), [10](#)

U

user_cache_dir() (in module homebase.homebase), [7](#)

user_config_dir() (in module homebase.homebase), [6](#)

user_data_dir() (in module homebase.homebase), [6](#)

user_logs_dir() (in module homebase.homebase), [8](#)

user_state_dir() (in module homebase.homebase), [8](#)

W

WINDOWS (Platform attribute), [6](#)