



Star 23,447

Dash Python > **Dash 1.0 Migration**

Plotly Studio: Transform any dataset into an interactive data application in minutes with AI. [Sign up for early access now.](#)

## Dash 1.0

Dash has been in active development for more than two years. In that time we've seen it evolve into a robust framework with a worldwide community of users. As we continued building Dash, talking and working with Dash users, we developed new features that obsoleted old ones, discovered which settings are awkward or confusing to new users, and overall accumulated a lot of ideas for how to improve the experience of building Dash apps.

A lot of these ideas have already been built into Dash. Sometimes they require changes that break backward compatibility. When these changes were small enough we added them anyway - we're still on major version 0 so **SemVer** allows breaking changes after all - but others we still held back because we expected them to impact too many users.

**Dash version 1.0**, in addition to various new features and upgrades we discuss elsewhere, collects those breaking changes all in one release. With it we are affirming the maturity and stability of the Dash and Dash component API. After making this migration, which we expect will impact a sizable fraction of Dash users in one way or another, Dash will not introduce any breaking changes unless and until we bump the major version again. That commitment includes the public API of the whole Dash "core": `dash`, `dash_core_components`, `dash_html_components`, and `dash_table`.

Here are the changes Dash developers need to know about to transition apps from v0.43 (the last release in the v0.x series) to v1.0.

### dash

#### **serve\_locally: default True and new kwarg**

Previously Dash would try to serve JavaScript and CSS files from public CDNs by default, and to disable this you needed to set:

```
app.scripts.config.serve_locally = True
app.css.config.serve_locally = True
```

Now that's the default, and that syntax can still be used to disable it for each asset class, but if you want to use CDNs for both you can just do:

```
app = dash.Dash(serve_locally=False)
```

Note that using `app.scripts.append_script` to load external js scripts won't work while `serve_locally=True`. Please use the `external_scripts` argument in your call to `dash.Dash` instead, as outlined in the [external resources section](#).

#### **Removed undo/redo buttons**

The undo & redo buttons in the corner of every app are removed by default, but if you want to re-enable them, set:



```
app = dash.Dash(show_undo_redo=True)
```

## Removed static\_folder

This old `dash.Dash` kwarg has been superseded by the much more powerful `assets_folder`. If you want the old behavior, use a custom `server`:

```
app = dash.Dash(server=Flask(static_folder='my_folder'))
```

## Removed components\_cache\_max\_age

All assets are served with cache-busting query strings in their URLs, so this is unnecessary.

## Use app.config

We used to store a lot of app settings directly as attributes, for example `app = dash.Dash(assets_ignore=...)` would set `app.assets_ignore`. Others would go into `app.config`. Now all constructor args go into `app.config`, with three exceptions:

- `server`: can set later with `app.init_app(server)`
- `index_string`: can set later with `app.index_string = 'custom string'`
- `plugins`: can add plugins later with `my_plugin.plugin(app)`

Some config items do not work correctly if modified later. These are read-only in `app.config`.

## Removed misspelled suppress\_callback\_exceptions

Just use the correct spelling, `suppress_callback_exceptions`

## Changed dev\_tools\_hot\_reload\_interval to seconds

Previously this `run_server` or `enable_dev_tools` setting was in milliseconds, but `dev_tools_hot_reload_watch_interval` is in seconds. Now they match!

## debug smarter defaults

Changed `enable_dev_tools()` to have `debug=True` by default. Now a simple `app.enable_dev_tools()` enables all the dev tools. In `app.run()` it is still `debug=False` by default, so `app.run(debug=True)` is still the correct way to enable dev tools there.

## dash\_core\_components

### Removed SyntaxHighlighter component

This is now built into `dcc.Markdown` using triple backticks, with optional short or long language name on the opening line:

```
dcc.Markdown('''
```py
def f(a, b):
    return a + b
```
''')
```



Added automatic dedenting in Markdown

If all lines (other than whitespace-only or blank lines) start with the same whitespace, it will be removed. Disable this with

```
dcc.Markdown(dedent=False)
```

Renamed `Checklist.values` prop to `value`

To match all the other input components.

Removed `version.py`

Until Dash v2, the simple way to get version info was `dcc.__version__`. Before Dash 1.0 there was also a `dash_core_components.version` module you could look in as well, but that has been removed. As of Dash v2, there is no longer a separate `dash_core_components` package at all. Now, the main `dash.__version__` is all you need to know.

dash\_html\_components

Removed `version.py`

Same as in `dash_core_components`. Until Dash v2, the simple way to get version info was `html.__version__`. Before Dash 1.0 there was also a `dash_html_components.version` module you could look in as well, but that has been removed. As of Dash v2, there is no longer a separate `dash_html_components` package at all. Now, the main `dash.__version__` is all you need to know.

dash\_table

Lots of properties are renamed and/or restructured for better self-consistency and extensibility.

| Old Prop         | New Prop            | Notes  |
|------------------|---------------------|--|
| column.clearable | dropdown*.clearable | clearable is a property of dropdowns, not really of columns. So it has moved into dropdown, dropdown_conditional, and dropdown_data.   |
| column.deletable |                     | For multi-line headers, accepts an array of booleans rather than a number. For example, if there are multiple headers and you want the second header row to be deletable, this would be <code>[False, True]</code> . |
| column.hidden    | REMOVED             | Will be added back later in a different form. Follow <a href="#">dash-table#314</a> for updates.   |
| column.id        |                     | Accepts strings only; numbers can be cast to string.   |



| Old Prop                                  | New Prop                                    | Notes  |
|---|---|--|
| <code>column.options</code>               | <code>dropdown</code>                       | Similar to <code>clearable</code> , <code>options</code> is a dropdown property, not a column property.  |
| <code>column.editable_name</code>         | <code>column.renamable</code>               | Similar to <code>deletable</code> , for multi-line headers, accepts an array of booleans rather than a number.   |
| <code>column_conditional_dropdowns</code> | <code>dropdown_conditional</code>           | Still an array of objects, but structure changed to <code>{clearable: boolean, if: {column_id, filter_query}, options: array of {label, value}}</code> .                     |
| <code>column_conditional_tooltips</code>  | <code>tooltip_conditional</code>            | Just renamed, structure is unchanged.  |
| <code>column_static_dropdown</code>       | <code>dropdown</code>                       | Changed from an array of objects to an object of objects. Keys are column IDs, values are <code>{clearable: boolean, options: array of {label, value}}</code> .              |
| <code>column_static_tooltip</code>        | <code>tooltip</code>                        | Renamed, values are unchanged.   |
| <code>content_style</code>                | <b>REMOVED</b>                              | Deemed unnecessary. NOTE - This was added back in 1.0.2 (dash-table==4.0.2) under the name the "fill_width" property name.   |
| <code>dropdown_properties</code>          | <code>dropdown_data</code>                  | Structure changed to match data: an array of objects, keys are column IDs, values are <code>{clearable: boolean, options: array of {label, value}}</code> .                  |
| <code>derived_filter_structure</code>     | <code>derived_filter_query_structure</code> | Just renamed, values are unchanged.  |
| <code>filter</code>                       | <code>filter_query</code>                   | Renamed, values are unchanged.   |
| <code>filtering</code>                    | <code>filter_action</code>                  | Values changed: 'fe' is now 'native', 'be' is now 'custom', and false is now 'none'  |
| <code>n_fixed_columns</code>              | <code>fixed_columns</code>                  | Now accepts an object <code>{headers: boolean, data: number}</code> instead of a number. <code>headers=True</code> fixes all row action columns (select/delete), data is the |



| Old Prop                           | New Prop                | Notes  |
|------------------------------------|-------------------------|--|
|                                    |                         | number of data columns to fix after the headers.   |
| n_fixed_rows                       | fixed_rows              | Now accepts an object {headers: boolean, data: number} instead of a number.<br>headers=True fixes all header rows, data is the number of data rows to fix after the headers. |
| pagination_mode                    | page_action             | Values changed: 'fe' is now 'native', 'be' is now 'custom', and false is now 'none'  |
| pagination_settings                | page_current, page_size | Split this nested property into two separate top-level properties.   |
| sort_type                          | sort_mode               | Renamed, values are unchanged.   |
| sorting                            | sort_action             | Values changed: 'fe' is now 'native', 'be' is now 'custom', and false is now 'none'  |
| sorting_treat_empty_string_as_none | sort_as_null            | Now accepts an array of string, number or booleans that can be ignored during sort. Can also be set per column with column.sort_as_null.                                     |
| style_data_conditional             |                         | Renamed nested prop if.filter to if.filter_query.  |
| tooltips                           | tooltip_data            | Structure changed to match data: an array of objects, keys are column IDs, values are {delay, duration, type, value} as before.  |

Additional notes

`style_cell_conditional`: In prior versions `if.row_index` was evaluated for this prop but should not have been. Use `style_data_conditional` and `style_header_conditional` instead for `if.row_index` based styling.

Dash Python > **Dash 1.0 Migration**

Products

Dash  
Consulting and Training

Pricing

Enterprise Pricing

About Us

Careers  
Resources

Support

Community Support  
Graphing Documentation

Join our mailing

list



Blog

Sign up to stay in the loop with all things Plotly — from Dash Club to product updates, webinars, and more!

**SUBSCRIBE**

Copyright © 2025 Plotly. All rights reserved.

[Terms of Service](#) [Privacy Policy](#)