

---

# **IPython Kernel Documentation**

*Release 4.7.0.dev*

**IPython Development Team**

**Apr 11, 2017**



---

# Contents

---

<b>1</b>	<b>Changes in IPython kernel</b>	<b>3</b>
1.1	4.6 . . . . .	3
1.2	4.5 . . . . .	4
1.3	4.4 . . . . .	4
1.4	4.3 . . . . .	5
1.5	4.2 . . . . .	5
1.6	4.1 . . . . .	6
1.7	4.0 . . . . .	6
<b>2</b>	<b>Indices and tables</b>	<b>7</b>



This contains minimal version-sensitive documentation for the IPython kernel package. Most IPython kernel documentation is in the [IPython documentation](#).

Contents:



---

## Changes in IPython kernel

---

### 4.6

#### 4.6.1

##### 4.6.1 on GitHub

- Fix eventloop-integration bug preventing Qt windows/widgets from displaying with ipykernel 4.6.0 and IPython 5.2.
- Avoid deprecation warnings about naive datetimes when working with `jupyter_client` 5.0.

#### 4.6.0

##### 4.6.0 on GitHub

- Add to `API DisplayPublisher.publish` two new fully backward-compatible keyword-args:
  - `update: bool`
  - `transient: dict`
- Support new `transient` key in `display_data` messages spec for `publish`. For a `display data` message, `transient` contains data that shouldn't be persisted to files or documents. Add a `display_id` to this `transient dict` by `display(obj, display_id=...)`
- Add `ipykernel_launcher` module which removes the current working directory from `sys.path` before launching the kernel. This helps to reduce the cases where the kernel won't start because there's a `random.py` (or similar) module in the current working directory.
- Add `busy/idle` messages on `IOPub` during processing of aborted requests
- Add `active event loop` setting to GUI, which enables the correct response to IPython's `is_event_loop_running_xxx`
- Include IPython kernelspec in wheels to reduce reliance on “native kernel spec” in `jupyter_client`

- Modify `OutputStream` to inherit from `TextIOBase` instead of `object` to improve API support and error reporting
- Fix IPython kernel death messages at start, such as “Kernel Restarting...” and “Kernel appears to have died”, when parent-poller handles PID 1
- Various bugfixes

## 4.5

### 4.5.2

[4.5.2 on GitHub](#)

- Fix bug when instantiating `Comms` outside of the IPython kernel (introduced in 4.5.1).

### 4.5.1

[4.5.1 on GitHub](#)

- Add missing `stream` parameter to overridden `getpass()`
- Remove locks from `iopub` thread, which could cause deadlocks during debugging
- Fix regression where `KeyboardInterrupt` was treated as an aborted request, rather than an error
- Allow instantiating `Comms` outside of the IPython kernel

### 4.5.0

[4.5 on GitHub](#)

- Use `figure.dpi` instead of `savefig.dpi` to set DPI for inline figures
- Support `ipympl` matplotlib backend (requires IPython update as well to fully work)
- Various bugfixes, including fixes for output coming from threads, and `input()` when called with non-string prompts, which `stdlib` allows.

## 4.4

### 4.4.1

[4.4.1 on GitHub](#)

- Fix circular import of `matplotlib` on Python 2 caused by the inline backend changes in 4.4.0.

### 4.4.0

[4.4.0 on GitHub](#)

- Use `MPLBACKEND` environment variable to tell `matplotlib`  $\geq 1.5$  use the inline backend by default. This is only done if `MPLBACKEND` is not already set and no backend has been explicitly loaded, so setting `MPLBACKEND=Qt4Agg` or calling `%matplotlib notebook` or `matplotlib.use('Agg')` will take precedence.



- Fixes for logging problems caused by 4.3, where logging could go to the terminal instead of the notebook.
- Add `--sys-prefix` and `--profile` arguments to `ipython kernel install`
- Allow Comm (Widget) messages to be sent from background threads.
- Select inline matplotlib backend by default if `%matplotlib magic` or `matplotlib.use()` are not called explicitly (for matplotlib  $\geq$  1.5).
- Fix some longstanding minor deviations from the message protocol (missing status: ok in a few replies, `connect_reply` format).
- Remove calls to `NoOpContext` from IPython, deprecated in 5.0.

## 4.3

### 4.3.2

- Use a nonempty dummy session key for inprocess kernels to avoid security warnings.

### 4.3.1

- Fix Windows Python 3.5 incompatibility caused by fault handler patch in 4.3

### 4.3.0

#### 4.3.0 on GitHub

- Publish all IO in a thread, via `IOPubThread`. This solves the problem of requiring `sys.stdout.flush()` to be called in the notebook to produce output promptly during long-running cells.
- Remove references to outdated IPython `gui` in kernel banner.
- Patch fault handler to use `sys.__stderr__` instead of forwarded `sys.stderr`, which has no `fileno` when forwarded.
- Deprecate some vestiges of the Big Split: - `ipykernel.find_connection_file()` is deprecated. Use `jupyter_client.find_connection_file()` instead. - Various pieces of code specific to IPython parallel are deprecated in `ipykernel` and moved to `ipyparallel`.

## 4.2

### 4.2.2

#### 4.2.2 on GitHub

- Don't show interactive debugging info when kernel crashes
- Fix handling of numerical types in `json_clean`
- Testing fixes for output capturing

## 4.2.1

[4.2.1 on GitHub](#)

- Fix default display name back to “Python X” instead of “pythonX”

## 4.2.0

[4.2 on GitHub](#)

- Support sending a full message in initial opening of comms (metadata, buffers were not previously allowed)
- When using `ipython kernel install --name` to install the IPython kernelspec, default `display-name` to the same value as `--name`.

## 4.1

### 4.1.1

[4.1.1 on GitHub](#)

- Fix missing `ipykernel.__version__` on Python 2.
- Fix missing `target_name` when opening comms from the frontend.

### 4.1.0

[4.1 on GitHub](#)

- add `ipython kernel install` entrypoint for installing the IPython kernelspec
- provisional implementation of `comm_info` request/reply for msgspec v5.1

## 4.0

[4.0 on GitHub](#)

4.0 is the first release of `ipykernel` as a standalone package.

## CHAPTER 2

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`