

# Contents

<b>1</b>	<b>Yaji Browser Side API</b>	<b>1</b>
1.1	Class Hierarchy	1
1.2	Class List	1
1.3	AbstractFileDialog Class Reference	2
1.3.1	Detailed Description	2
1.4	Curtain Class Reference	3
1.4.1	Detailed Description	3
1.5	Dialog Class Reference	3
1.5.1	Detailed Description	3
1.6	OpenDirectoryDialog Class Reference	3
1.6.1	Detailed Description	4
1.7	OpenFileDialog Class Reference	4
1.7.1	Detailed Description	4
1.8	PyHtmlViewSupport Class Reference	4
1.8.1	Detailed Description	5
1.9	TrStr Class Reference	5
1.9.1	Detailed Description	5
1.10	YajiAjaxError Class Reference	5
1.10.1	Detailed Description	5
1.11	YajiAjaxRequest Class Reference	5
1.11.1	Detailed Description	6
1.12	YajiClient Class Reference	6
1.12.1	Detailed Description	7

1.12.2	Member Function Documentation	7
1.12.2.1	addEventHandler()	7
1.12.2.2	ajax()	7
1.12.2.3	getClientlocalDatastore()	7
1.12.2.4	getDatastore()	8
1.12.2.5	ready()	8
1.12.2.6	stop()	8
1.12.2.7	tryJsonStringify()	9
1.13	YajiClientEvents Class Reference	9
1.13.1	Detailed Description	9
1.14	YajiConfiguration Class Reference	9
1.14.1	Detailed Description	10
1.14.2	Member Function Documentation	10
1.14.2.1	addHandler()	10
1.14.2.2	getConfig()	10
1.14.2.3	setConfig()	10
1.15	YajiDataStore Class Reference	11
1.15.1	Detailed Description	11
1.16	YajiDialogs Class Reference	11
1.16.1	Detailed Description	11
1.17	YajiPopupMenuButton Class Reference	11
1.17.1	Detailed Description	12
1.18	YajiUserFeedback Class Reference	12
1.18.1	Detailed Description	12
1.18.2	Member Function Documentation	12
1.18.2.1	registerHandler()	12
1.18.2.2	show()	12

# Chapter 1

## Yaji Browser Side API

This is the API reference for the browser side (JavaScript) part of Yaji. For general information, also about many browser side topics, please read README.pdf before!

Inside this file, you might want to start with [YajiClient](#).

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

AbstractFileDialog . . . . .	2
OpenDirectoryDialog . . . . .	3
OpenFileDialog . . . . .	4
Curtain . . . . .	3
Dialog . . . . .	3
PyHtmlViewSupport . . . . .	4
TrStr . . . . .	5
YajiAjaxError . . . . .	5
YajiAjaxRequest . . . . .	5
YajiClient . . . . .	6
YajiClientEvents . . . . .	9
YajiConfiguration . . . . .	9
YajiDataStore . . . . .	11
YajiDialogs . . . . .	11
YajiPopupMenuButton . . . . .	11
YajiUserFeedback . . . . .	12

### 1.2 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

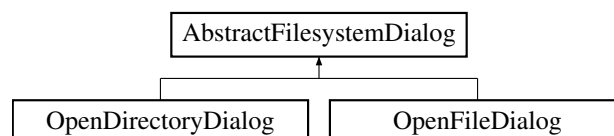
<a href="#">AbstractFileDialog</a>	
Abstract filesystem picker dialog builder. See subclasses . . . . .	2
<a href="#">Curtain</a>	
User interface curtains are used for disabling parts of the user interface during some backend operations . . . . .	3

<a href="#">Dialog</a>	A dialog (either modal or non-modal).	3
<a href="#">OpenDirectoryDialog</a>	Open directory dialog builder	3
<a href="#">OpenFileDialog</a>	Open file dialog builder	4
<a href="#">PyHtmlViewSupport</a>	Browser side support for PyHtmlView	4
<a href="#">TrStr</a>	An i18n-aware (translatable) string	5
<a href="#">YajiAjaxError</a>	Backend errors in answering ajax requests	5
<a href="#">YajiAjaxRequest</a>	Handle object for an ajax request (triggered by <a href="#">YajiClient.ajax()</a> )	5
<a href="#">YajiClient</a>	The application main object. Usually you can access it by just <code>yaji</code>	6
<a href="#">YajiClientEvents</a>	Internal handling of client events. See <a href="#">YajiClient.addEventHandler()</a> .	9
<a href="#">YajiConfiguration</a>	The application configuration. See also <code>app.Application.appconfig()</code>	9
<a href="#">YajiDataStore</a>	A special <code>clove::AjaxAsyncDatasource</code> backed by a datasource on application code side. See also <code>app.Application.create_datastore()</code> and <code>app.Application.bindserver()</code>	11
<a href="#">YajiDialogs</a>	Manager for backend controlled dialogs	11
<a href="#">YajiPopupMenuButton</a>	A special <code>clove::PopupMenuButton</code> connected to application's action execution framework.	11
<a href="#">YajiUserFeedback</a>	User feedback manager. This is used for simple dialogs like message boxes, input boxes, and so on	12

## 1.3 AbstractFileDialog Class Reference

Abstract filesystem picker dialog builder. See subclasses.

Inheritance diagram for AbstractFileDialog:



### Public Member Functions

- [showDialog\(\)](#)  
*Shows the dialog and calls either `onsuccess` or `oncancel`.*

#### 1.3.1 Detailed Description

Abstract filesystem picker dialog builder. See subclasses.

The documentation for this class was generated from the following file:

- `/home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js`

## 1.4 Curtain Class Reference

User interface curtains are used for disabling parts of the user interface during some backend operations.

### Public Member Functions

- `close ()`  
*Implements closing the curtain, so blocking some parts of the user interface in some ways. If you override it, also override `open()`.*
- `open ()`  
*Implements opening the curtain, so unblocking the user interface again. If you override it, also override `close()`.*
- `async execute (fct)`  
*Executes a function inside the curtain, so the curtain gets closed before, and gets opened when processing is idle again.*

### 1.4.1 Detailed Description

User interface curtains are used for disabling parts of the user interface during some backend operations.

The default implementation disables the entire user interface with a modal busy animation for that, so in many cases you have to subclass an own implementation.

The documentation for this class was generated from the following file:

- `/home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js`

## 1.5 Dialog Class Reference

A dialog (either modal or non-modal). .

### 1.5.1 Detailed Description

A dialog (either modal or non-modal). .

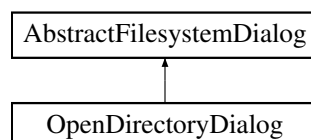
The documentation for this class was generated from the following file:

- `/home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js`

## 1.6 OpenFileDialog Class Reference

Open directory dialog builder.

Inheritance diagram for OpenFileDialog:



## Public Member Functions

- [showDialog \(\)](#)  
*Shows the dialog and calls either `onsuccess` or `oncancel`.*

### 1.6.1 Detailed Description

Open directory dialog builder.

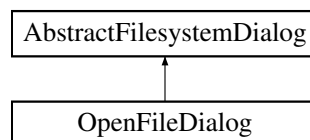
The documentation for this class was generated from the following file:

- `/home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js`

## 1.7 OpenFileDialog Class Reference

Open file dialog builder.

Inheritance diagram for OpenFileDialog:



## Public Member Functions

- [showDialog \(\)](#)  
*Shows the dialog and calls either `onsuccess` or `oncancel`.*

### 1.7.1 Detailed Description

Open file dialog builder.

The documentation for this class was generated from the following file:

- `/home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js`

## 1.8 PyHtmlViewSupport Class Reference

Browser side support for PyHtmlView.

## Public Member Functions

- [constructor](#) (app)
- [\\_onwindowclose](#) ()

*Used by PyHtmlView when the user tries to close the window.*

### 1.8.1 Detailed Description

Browser side support for PyHtmlView.

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.9 TrStr Class Reference

An i18n-aware (translatable) string.

### 1.9.1 Detailed Description

An i18n-aware (translatable) string.

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.10 YajiAjaxError Class Reference

Backend errors in answering ajax requests.

Inherits Error.

### 1.10.1 Detailed Description

Backend errors in answering ajax requests.

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.11 YajiAjaxRequest Class Reference

Handle object for an ajax request (triggered by [YajiClient.ajax\(\)](#)).

### 1.11.1 Detailed Description

Handle object for an ajax request (triggered by [YajiClient.ajax\(\)](#)).

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.12 YajiClient Class Reference

The application main object. Usually you can access it by just `yaji`.

### Public Member Functions

- [get appconfig](#) ()  
*Instance of [YajiConfiguration](#).*
- [get userfeedback](#) ()  
*Instance of [YajiUserFeedback](#).*
- [get pyhtmlview](#) ()  
*Instance of [PyHtmlViewSupport](#).*
- [get mainview](#) ()  
*Instance of [clove::MainView](#).*
- [get isrunning](#) ()  
*If the application is currently running.*
- [get hasExternalCloseControls](#) ()  
*If this client view has dedicated external close controls (e.g. via [PyHtmlView](#)).*
- [get isStaticApplication](#) ()  
*If the application is static. See [app.Application.storeasstaticapplication\(\)](#).*
- [get OnUiShutdown](#) ()  
*A [clove::Event](#) which is trigger on user interface shutdown. See also [YajiClient.uiShutdown\(\)](#).*
- [addEventListener](#) (key, fct)  
*Adds a handler function for a backend event. See [app.Application.triggerevent\(\)](#).*
- [tryJsonStringify](#) (o)  
*Returns a json representation for an object, just containing primitive values, arrays and dicts.*
- [stop](#) (forceBlockUiDuringRequest)  
*Completely stops the application, including application code side. See [app.Application.stop\(\)](#) for explanation about stop procedure, and also [YajiClient.uiShutdown\(\)](#).*
- [ajax](#) (config)  
*Makes an ajax request.*
- [async getDatastore](#) (name)  
*Returns a application code side datasource by name. See also [app.Application.create\\_datastore\(\)](#) and [app.<← Application.binddata\(\)](#).*
- [getClientlocalDatastore](#) (name)  
*Returns a browser side datasource by name. See also [app.Application.create\\_clientlocal\\_datastore\(\)](#) and [app.<← Application.binddata\(\)](#).*
- [uiShutdown](#) ()  
*Shuts down the user interface (without killing the backend). See also [YajiClient.stop\(\)](#).*
- [ready](#) (fct)  
*Executes a function on startup once the yaji app is initialized.*



### 1.12.1 Detailed Description

The application main object. Usually you can access it by just `yaji`.

### 1.12.2 Member Function Documentation

#### 1.12.2.1 addEventHandler()

```
YajiClient::addEventHandler (
    key,
    fct )
```

Adds a handler function for a backend event. See `app.Application.triggerevent()`.

##### Parameters

<i>key</i>	The event key (or event 'name').
<i>fct</i>	The handler function. Receives the event data as first argument.

#### 1.12.2.2 ajax()

```
YajiClient::ajax (
    config )
```

Makes an ajax request.

It returns a Promise-like object, so it can be `awaited`.

`config` may contain the same as for `$.ajax`, and also:

- `finished`: Callback which gets executed after either success or error.
- `alsoAfterShutdown`: If to make that request even if the ui is already stopped.

##### Parameters

<i>config</i>	The request configuration.
---------------	----------------------------

#### 1.12.2.3 getClientlocalDatastore()

```
YajiClient::getClientlocalDatastore (
    name )
```

Returns a browser side datasource by name. See also `app.Application.create_clientlocal_datastore()` and `app.Application.binddata()`.

#### Parameters

<i>name</i>	The datasource name.
-------------	----------------------

#### 1.12.2.4 `getDatastore()`

```
async YajiClient::getDatastore (
    name )
```

Returns a application code side datasource by name. See also `app.Application.create_datastore()` and `app.Application.binddata()`.

#### Parameters

<i>name</i>	The datasource name.
-------------	----------------------

#### 1.12.2.5 `ready()`

```
YajiClient::ready (
    fct )
```

Executes a function on startup once the yaji app is initialized.

#### Parameters

<i>fct</i>	The function to execute.
------------	--------------------------

#### 1.12.2.6 `stop()`

```
YajiClient::stop (
    forceBlockUiDuringRequest )
```

Completely stops the application, including application code side. See `app.Application.stop()` for explanation about stop procedure, and also [YajiClient.uiShutdown\(\)](#).

#### Parameters

<i>forceBlockUiDuringRequest</i>	If to force freezing the user interface by a modal panel (which automatically happens only in some situations) during stop request.
----------------------------------	---

### 1.12.2.7 tryJsonStringify()

```
YajiClient::tryJsonStringify (
    o )
```

Returns a json representation for an object, just containing primitive values, arrays and dicts.

#### Parameters

<i>o</i>	The object to serialize.
----------	--------------------------

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.13 YajiClientEvents Class Reference

Internal handling of client events. See [YajiClient.addEventHandler\(\)](#). .

### 1.13.1 Detailed Description

Internal handling of client events. See [YajiClient.addEventHandler\(\)](#). .

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.14 YajiConfiguration Class Reference

The application configuration. See also `app.Application.appconfig()`.

### Public Member Functions

- [addHandler](#) (key, fct, runAlsoOnNeutralAssignments)  
*Adds a configuration handler for a config key. It gets called whenever the value for this key changes, and also directly if there already is something stored for that key at execution time.*
- [getConfig](#) (key)  
*Returns a configuration value by key.*
- [setConfig](#) (key, value)  
*Requests to set a configuration value for a key. Note: This happens asynchronously by a request to the backend.*

### 1.14.1 Detailed Description

The application configuration. See also `app.Application.appconfig()`.

### 1.14.2 Member Function Documentation

#### 1.14.2.1 `addHandler()`

```
YajiConfiguration::addHandler (
    key,
    fct,
    runAlsoOnNeutralAssignments )
```

Adds a configuration handler for a config key. It gets called whenever the value for this key changes, and also directly if there already is something stored for that key at execution time.

##### Parameters

<i>key</i>	The configuration key.
<i>fct</i>	The handler function.
<i>runAlsoOnNeutralAssignments</i>	If to run it also for assignments which haven't changed the value.

#### 1.14.2.2 `getConfig()`

```
YajiConfiguration::getConfig (
    key )
```

Returns a configuration value by key.

##### Parameters

<i>key</i>	The configuration key.
------------	------------------------

#### 1.14.2.3 `setConfig()`

```
YajiConfiguration::setConfig (
    key,
    value )
```

Requests to set a configuration value for a key. Note: This happens asynchronously by a request to the backend.

**Parameters**

<i>key</i>	The configuration key.
<i>value</i>	The new configuration value.

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.15 YajiDataStore Class Reference

A special clove::AjaxAsyncDatasource backed by a datasource on application code side. See also [app.Application.create\\_datastore\(\)](#) and [app.Application.bindserver\(\)](#).

Inherits AjaxAsyncDatasource.

### 1.15.1 Detailed Description

A special clove::AjaxAsyncDatasource backed by a datasource on application code side. See also [app.Application.create\\_datastore\(\)](#) and [app.Application.bindserver\(\)](#).

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.16 YajiDialogs Class Reference

Manager for backend controlled dialogs.

### 1.16.1 Detailed Description

Manager for backend controlled dialogs.

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.17 YajiPopupMenuButton Class Reference

A special clove::PopupMenuButton connected to application's action execution framework. .

Inherits PopupMenuButton.

### 1.17.1 Detailed Description

A special clove::PopupMenuButton connected to application's action execution framework. .

The documentation for this class was generated from the following file:

- /home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js

## 1.18 YajiUserFeedback Class Reference

User feedback manager. This is used for simple dialogs like message boxes, input boxes, and so on.

### Public Member Functions

- [show](#) (feedbackcfg, onanswered)
- [registerHandler](#) (kind, fct)

*Registers a handler for a custom kind of user feedback dialog.*

### 1.18.1 Detailed Description

User feedback manager. This is used for simple dialogs like message boxes, input boxes, and so on.

### 1.18.2 Member Function Documentation

#### 1.18.2.1 registerHandler()

```
YajiUserFeedback::registerHandler (
    kind,
    fct )
```

Registers a handler for a custom kind of user feedback dialog.

#### Parameters

<i>kind</i>	The custom dialog kind name.
<i>fct</i>	The custom handler function for this dialog kind.

#### 1.18.2.2 show()

```
YajiUserFeedback::show (
```

```
feedbackcfg,  
onanswered )
```

Shows a feedback dialog by feedback configuration and calls a callback for the answer.

#### Parameters

<i>feedbackcfg</i>	Feedback configuration, incl. <code>kind</code> and some kind-dependent parameters.
<i>onanswered</i>	The callback for the dialog answer.

The documentation for this class was generated from the following file:

- `/home/pino/projects/anise/anise/data/Yaji/yaji/static/yaji.js`





# Index

- AbstractFileDialog, [2](#)
- addEventHandler
  - YajiClient, [7](#)
- addHandler
  - YajiConfiguration, [10](#)
- ajax
  - YajiClient, [7](#)
- Curtain, [3](#)
- Dialog, [3](#)
- getClientlocalDatastore
  - YajiClient, [7](#)
- getConfig
  - YajiConfiguration, [10](#)
- getDatastore
  - YajiClient, [8](#)
- OpenDirectoryDialog, [3](#)
- OpenFileDialog, [4](#)
- PyHtmlViewSupport, [4](#)
- ready
  - YajiClient, [8](#)
- registerHandler
  - YajiUserFeedback, [12](#)
- setConfig
  - YajiConfiguration, [10](#)
- show
  - YajiUserFeedback, [12](#)
- stop
  - YajiClient, [8](#)
- TrStr, [5](#)
- tryJsonStringify
  - YajiClient, [9](#)
- YajiAjaxError, [5](#)
- YajiAjaxRequest, [5](#)
- YajiClient, [6](#)
  - addEventHandler, [7](#)
  - ajax, [7](#)
  - getClientlocalDatastore, [7](#)
  - getDatastore, [8](#)
  - ready, [8](#)
  - stop, [8](#)
  - tryJsonStringify, [9](#)
- YajiClientEvents, [9](#)
- YajiConfiguration, [9](#)
  - addHandler, [10](#)
  - getConfig, [10](#)
  - setConfig, [10](#)
- YajiDataStore, [11](#)
- YajiDialogs, [11](#)
- YajiPopupMenuButton, [11](#)
- YajiUserFeedback, [12](#)
  - registerHandler, [12](#)
  - show, [12](#)