

---

# **ats\_utilities Documentation**

*Release*

***[https://github.com/vroncevic/ats\\_utilities/releases](https://github.com/vroncevic/ats_utilities/releases)***

**Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)>**

**Dec 19, 2021**



---

## Contents:

---

<b>1</b>	<b>ats_utilities</b>	<b>3</b>
1.1	ats_utilities package . . . . .	3
1.1.1	Subpackages . . . . .	3
1.1.1.1	ats_utilities.abstract package . . . . .	3
1.1.1.2	ats_utilities.checker package . . . . .	4
1.1.1.3	ats_utilities.cli package . . . . .	6
1.1.1.4	ats_utilities.config_io package . . . . .	11
1.1.1.5	ats_utilities.console_io package . . . . .	27
1.1.1.6	ats_utilities.cooperative package . . . . .	31
1.1.1.7	ats_utilities.exceptions package . . . . .	32
1.1.1.8	ats_utilities.final package . . . . .	36
1.1.1.9	ats_utilities.info package . . . . .	37
1.1.1.10	ats_utilities.logging package . . . . .	42
1.1.1.11	ats_utilities.option package . . . . .	45
1.1.1.12	ats_utilities.register package . . . . .	46
1.1.1.13	ats_utilities.singleton package . . . . .	47
1.1.1.14	ats_utilities.splash package . . . . .	49
1.1.2	Module contents . . . . .	55
<b>2</b>	<b>Installation</b>	<b>57</b>
<b>3</b>	<b>Dependencies</b>	<b>59</b>
<b>4</b>	<b>Framework structure</b>	<b>61</b>
<b>5</b>	<b>Copyright and licence</b>	<b>65</b>
<b>6</b>	<b>Indices and tables</b>	<b>67</b>
	<b>Python Module Index</b>	<b>69</b>
	<b>Index</b>	<b>71</b>



**ats\_utilities** is framework for creating Apps/Tools/Scripts.

Developed in `python` code: **100%**.

The README is used to introduce the modules and provide instructions on how to install the modules, any machine dependencies it may have and any other information that should be provided before the modules are installed.



## 1.1 ats\_utilities package

### 1.1.1 Subpackages

#### 1.1.1.1 ats\_utilities.abstract package

##### Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `AbstractMethod` with attribute(s) and method(s). Created abstract decorator for object methods.

**class** `ats_utilities.abstract.AbstractMethod` (*method\_to\_abstract*)  
Bases: `object`

Defined class `AbstractMethod` with attribute(s) and method(s). Created abstract decorator for object methods.  
It defines:

##### attributes

- `method_name` - object method name.
- `method_class_name` - class name of method.
- `method_type` - type of object method.
- `method` - method object-reference.

### methods

`__init__` - initial constructor.  
`__call__` - raise exception `NotImplementedError` on call.  
`__str__` - str dunder method for `AbstractMethod`.

## 1.1.1.2 ats\_utilities.checker package

### Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSChecker` with attribute(s) and method(s). Created API fo checking parameters for object methods and functions.

**class** `ats_utilities.checker.ATSChecker`

Bases: `object`

Defined class `ATSChecker` with attribute(s) and method(s). Created API fo checking parameters for object methods and functions. It defines:

### attributes

`NO_ERROR` - no error, error id (0).  
`TYPE_ERROR` - type param error id (1).  
`VALUE_ERROR` - value param error id (2).  
`FORMAT_ERROR` - wrong format error id (3).  
`__start_message` - start segment of usage message.  
`__list_of_params` - list of parameters for method/function.  
`__error_type` - list of mapped errors.  
`__error_type_index` - error type index.  
`__error_value_index` - error value index.

### methods

`__init__` - initial constructor.  
`collect_params` - collect all parameters in one list.  
`usage_message` - prepare usage message for method/function.  
`check_types` - check parameters (types) for method/function.  
`check_values` - check parameters (values) for method/function.  
`priority_error` - set priority error id (`TYPE_ERROR`).  
`check_params` - check parameters for method/function.  
`__str__` - str dunder method for object `ATSChecker`.

`FORMAT_ERROR = 3`

`NO_ERROR = 0`



**TYPE\_ERROR = 1**

**VALUE\_ERROR = 2**

**check\_params** (*params\_description*)

Check parameters for method/function.

**Parameters** **params\_description** (<list>) – parameters description.

**Returns** usage message, status (0 | 1 | 2 | 3).

**Return type** <str>, <int>

**Exceptions** None

**check\_types** (*params\_description*)

Check parameters (types) for method/function.

**Parameters** **params\_description** (<OrderedDict>) – parameters description.

**Returns** boolean status, True (ok) | False.

**Return type** <bool>

**Exceptions** None

**check\_values** (*params\_description*)

Check parameters (values) for method/function.

**Parameters** **params\_description** (<OrderedDict>) – parameters description.

**Returns** boolean status, True (ok) | False.

**Return type** <bool>

**Exceptions** None

**collect\_params** (*params\_description*)

Collect all parameters in one list.

**Parameters**

- **params\_description** (<OrderedDict>) – description for parameters.
- **verbose** (<bool>) – enable/disable verbose option.

**Returns** True (format for params ok) | False.

**Returns** boolean status, True (ok) | False.

**Return type** <bool>

**Exceptions** None

**priority\_error** ()

Set priority error id (TYPE\_ERROR).

**Returns** priority error id (0 | 1 | 2 | 3) | None.

**Return type** <int> | <NoneType>

**Exceptions** None

**usage\_message** ()

Prepare usage message for method/function.

**Returns** usage message for method/function.

**Return type** <str>

**Exceptions** None

### 1.1.1.3 ats\_utilities.cli package

#### Submodules

#### ats\_utilities.cli.cfg\_cli module

**Module** `cfg_cli.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `CfgCLI` with attribute(s) and method(s). Created API for check and load informations, setup argument parser.

**class** `ats_utilities.cli.cfg_cli.CfgCLI` (*informations\_file, verbose=False*)  
Bases: `ats_utilities.config_io.cfg.CfgBase`

Defined class `CfgCLI` with attribute(s) and method(s). Created API for check and load informations, setup arguments parser. It defines:

#### attributes

`__verbose` - enable/disable verbose option.

#### methods

`__init__` - initial constructor.

`add_new_option` - adding new option for CL interface.

`parse_args` - parse arguments.

`process` - process and run tool operation (Abstract method).

`__str__` - str dunder method for `CfgCLI`.

**VERBOSE** = `'ats_utilities::cli::cfg_cli'`

**add\_new\_option** (*\*args, \*\*kwargs*)  
Adding new option for CL interface.

#### Parameters

- **args** (<list>) – list of arguments (objects).
- **kwargs** (<dict>) – arguments in shape of dictionary.

**Exceptions** None

**parse\_args** (*argv*)  
Process arguments from start.

**Parameters** **argv** (<list>) – arguments.

**Returns** options and arguments.

**Return type** <Python object(s)>

**Exceptions** None

```
process = <ats_utilities.abstract.AbstractMethod object>
```

## ats\_utilities.cli.ini\_cli module

**Module** ini\_cli.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class IniCLI with attribute(s) and method(s). Created API for check and load informations, setup argument parser.

**class** ats\_utilities.cli.ini\_cli.IniCLI (*informations\_file, verbose=False*)  
Bases: *ats\_utilities.config\_io.ini.IniBase*

Defined class IniCLI with attribute(s) and method(s). Created API for check and load informations, setup argument parser. It defines:

### attributes

\_\_verbose - enable/disable verbose option.

### methods

\_\_init\_\_ - initial constructor.

add\_new\_option - adding new option for CL interface.

parse\_args - parse arguments.

process - process and run tool operation (Abstract method).

\_\_str\_\_ - str dunder method for IniCLI.

```
VERBOSE = 'ats_utilities::cli::ini_cli'
```

```
add_new_option (*args, **kwargs)  
Adding new option for CL interface.
```

### Parameters

- **args** (<list>) – list of arguments (objects).
- **kwargs** (<dict>) – arguments in shape of dictionary.

**Exceptions** None

```
parse_args (argv)  
Process arguments from start.
```

**Parameters** **argv** (<list>) – arguments.

**Returns** options and arguments.

**Return type** <Python object(s)>

**Exceptions** None

```
process = <ats_utilities.abstract.AbstractMethod object>
```

## ats\_utilities.cli.json\_cli module

Module json\_cli.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class JsonCLI with attribute(s) and method(s). Created API for check and load informations, setup argument parser.

**class** ats\_utilities.cli.json\_cli.JsonCLI (*informations\_file, verbose=False*)  
Bases: *ats\_utilities.config\_io.json.JsonBase*

Defined class JsonCLI with attribute(s) and method(s). Created API for check and load informations, setup argument parser. It defines:

### attributes

\_\_verbose - enable/disable verbose option.

### methods

\_\_init\_\_ - initial constructor.

add\_new\_option - adding new option for CL interface.

parse\_args - parse arguments.

process - process and run tool operation (Abstract method).

\_\_str\_\_ - str dunder method for JsonCLI.

**VERBOSE** = 'ats\_utilities::cli::json\_cli'

**add\_new\_option** (*\*args, \*\*kwargs*)  
Adding new option for CL interface.

### Parameters

- **args** (<list>) – list of arguments (objects).
- **kwargs** (<dict>) – arguments in shape of dictionary.

**Exceptions** None

**parse\_args** (*argv*)  
Process arguments from start.

**Parameters** **argv** (<list>) – arguments.

**Returns** options and arguments.

**Return type** <Python object(s)>

**Exceptions** None

**process** = <ats\_utilities.abstract.AbstractMethod object>

## ats\_utilities.cli.xml\_cli module

Module xml\_cli.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class XmlCLI with attribute(s) and method(s). Created API for check and load informations, setup argument parser.

**class** `ats_utilities.cli.xml_cli.XmlCLI` (*informations\_file, verbose=False*)  
Bases: `ats_utilities.config_io.xml.XmlBase`

Defined class XmlCLI with attribute(s) and method(s). Created API for check and load informations, setup argument parser. It defines:

**attributes**

`__verbose` - enable/disable verbose option.

**methods**

`__init__` - initial constructor.

`add_new_option` - adding new option for CL interface.

`parse_args` - parse arguments.

`process` - process and run tool operation (Abstract method).

`__str__` - str dunder method for XmlCLI.

**VERBOSE** = `'ats_utilities::cli::xml_cli'`

**add\_new\_option** (*\*args, \*\*kwargs*)  
Adding new option for CL interface.

**Parameters**

- **args** (<list>) – list of arguments (objects).
- **kwargs** (<dict>) – arguments in shape of dictionary.

**Exceptions** None

**parse\_args** (*argv*)  
Process arguments from start.

**Parameters** **argv** (<list>) – arguments.

**Returns** options and arguments.

**Return type** <Python object(s)>

**Exceptions** None

**process** = <ats\_utilities.abstract.AbstractMethod object>

## ats\_utilities.cli.yaml\_cli module

**Module** `yaml_cli.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities

is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `YamlCLI` with attribute(s) and method(s). Created API for check and load informations, setup argument parser.

**class** `ats_utilities.cli.yaml_cli.YamlCLI` (*informations\_file, verbose=False*)  
Bases: `ats_utilities.config_io.yaml.YamlBase`

Defined class `YamlCLI` with attribute(s) and method(s). Created API for check and load informations, setup argument parser. It defines:

#### attributes

`__verbose` - enable/disable verbose option.

#### methods

`__init__` - initial constructor.

`add_new_option` - adding new option for CL interface.

`parse_args` - parse arguments.

`process` - process and run tool operation (Abstract method).

`__str__` - str dunder method for `YamlCLI`.

**VERBOSE** = `'ats_utilities::cli::yaml_cli'`

**add\_new\_option** (*\*args, \*\*kwargs*)  
Adding new option for CL interface.

#### Parameters

- **args** (*<list>*) – list of arguments (objects).
- **kwargs** (*<dict>*) – arguments in shape of dictionary.

**Exceptions** None

**parse\_args** (*argv*)  
Process arguments from start.

**Parameters** **argv** (*<list>*) – arguments.

**Returns** options and arguments.

**Return type** *<Python object(s)>*

**Exceptions** None

**process** = *<ats\_utilities.abstract.AbstractMethod object>*

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic [elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com) `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General

Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined package `ats_utilities.cli`.

#### 1.1.1.4 `ats_utilities.config_io` package

##### Subpackages

##### `ats_utilities.config_io.cfg` package

##### Submodules

##### `ats_utilities.config_io.cfg.cfg2object` module

**Module** `cfg2object.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `Cfg2Object` with attribute(s) and method(s). Created API for reading configuration/information from a `cfg` file.

**class** `ats_utilities.config_io.cfg.cfg2object.Cfg2Object` (*configuration\_file*, *verbose=False*)

Bases: `ats_utilities.config_io.base_read.BaseReadConfig`

Defined class `Cfg2Object` with attribute(s) and method(s). Created API for read configuration/information from a `cfg` file. It defines:

##### attributes

`__FORMAT` - format of configuration content.  
`__REGEX_MATCH_LINE` - regular expression for matching line.  
`__verbose` - enable/disable verbose option.

##### methods

`__init__` - initial constructor.  
`read_configuration` - read configuration from file.  
`__str__` - str dunder method for object `Cfg2Object`.

`VERBOSE = 'ats_utilities::config_io::cfg::cfg2object'`

`_Cfg2Object__FORMAT = 'cfg'`

`_Cfg2Object__REGEX_MATCH_LINE = '^\\s*$'`

`read_configuration` (*verbose=False*)

Getting a configuration from `cfg` file.

**Parameters** `verbose` (<*bool*>) – enable/disable verbose option.

**Returns** configuration object | None.

**Return type** <dict> | <NoneType>

**Exceptions** None

## ats\_utilities.config\_io.cfg.object2cfg module

**Module** object2cfg.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Object2Cfg with attribute(s) and method(s). Created API for writing configuration/information to a cfg file.

**class** ats\_utilities.config\_io.cfg.object2cfg.Object2Cfg(*configuration\_file*, *verbose=False*)

Bases: *ats\_utilities.config\_io.base\_write.BaseWriteConfig*

Defined class Object2Cfg with attribute(s) and method(s). Created API for writing configuration/information to a cfg file. It defines:

### attributes

`__FORMAT` - format of configuration content.

`__verbose` - enable/disable verbose option.

### methods

`__init__` - initial constructor.

`write_configuration` - write config to a cfg file.

`__str__` - str dunder method for object Object2Cfg.

`VERBOSE = 'ats_utilities::config_io::cfg::object2cfg'`

`__Object2Cfg__FORMAT = 'cfg'`

`write_configuration`(*configuration*, *verbose=False*)

Write configuration to a cfg file.

### Parameters

- **configuration** (<Python object(s)> | <NoneType>) – configuration object | None.
- **verbose** (<bool>) – enable/disable verbose option.

**Returns** boolean status, True (success) | False.

**Return type** <bool>

**Exceptions** None

## Module contents

**Module** `__init__.py`



**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class CfgBase with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface.

**class** `ats_utilities.config_io.cfg.CfgBase` (*informations\_file, verbose=False*)  
Bases: object

Defined class CfgBase with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface. It defines:

**attributes**

`__verbose` - enable/disable verbose option.  
`tool_operational` - control ATS operational functionality.  
`cfg2obj` - in API for informations.  
`obj2cfg` - out API for informations.  
`option_parser` - option parser for ATS.

**methods**

`__init__` - initial constructor.  
`is_tool_ok` - checking is tool operational.  
`__str__` - str dunder method for object CfgBase.

**VERBOSE** = `'ats_utilities::config_io::cfg'`

**is\_tool\_ok** ()

Checking is tool operational.

**Returns** boolean status, True (yes) | False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.config\_io.ini package

### Submodules

#### ats\_utilities.config\_io.ini.ini2object module

**Module** ini2object.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Ini2Object with attribute(s) and method(s). Created API for reading configuration/information from an ini file.

```
class ats_utilities.config_io.ini.ini2object.Ini2Object (configuration_file, verbose=False)
Bases: ats_utilities.config_io.base_read.BaseReadConfig
```

Defined class Ini2Object with attribute(s) and method(s). Created API for reading configuration/information from an ini file. It defines:

**attributes**

\_\_FORMAT - format of configuration content.  
\_\_verbose - enable/disable verbose option.

**methods**

\_\_init\_\_ - initial constructor.  
read\_configuration - read configuration from file.  
\_\_str\_\_ - str dunder method for object Ini2Object.

```
VERBOSE = 'ats_utilities::config_io::ini::ini2object'
```

```
_Ini2Object__FORMAT = 'ini'
```

```
read_configuration (verbose=False)
Getting a configuration from an ini file.
```

**Parameters** *verbose* (<bool>) – enable/disable verbose option.

**Returns** configuration object | None.

**Return type** <ConfigParser> | <NoneType>

## ats\_utilities.config\_io.ini.object2ini module

**Module** object2ini.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Object2Ini with attribute(s) and method(s). Created API for writing configuration/information to an ini file.

```
class ats_utilities.config_io.ini.object2ini.Object2Ini (configuration_file, verbose=False)
Bases: ats_utilities.config_io.base_write.BaseWriteConfig
```

Defined class Object2Ini with attribute(s) and method(s). Created API for writing configuration/information to an ini file. It defines:

**attributes**

\_\_FORMAT - format of configuration content.  
\_\_verbose - enable/disable verbose option.

**methods**

`__init__` - initial constructor.  
`write_configuration` - write configuration to an ini file.  
`__str__` - str dunder method for object `Object2Ini`.

**VERBOSE** = 'ats\_utilities::config\_io::ini::object2ini'

**\_Object2Ini\_\_FORMAT** = 'ini'

**write\_configuration** (*configuration*, *verbose=False*)  
Write configuration to a ini file.

#### Parameters

- **configuration** (<Python object(s)> | <NoneType>) – configuration object | None.
- **verbose** (<bool>) – enable/disable verbose option.

**Returns** boolean status, True (success) | False.

**Return type** <bool>

**Exception** None

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `IniBase` with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface.

**class** `ats_utilities.config_io.ini.IniBase` (*informations\_file*, *verbose=False*)  
Bases: `object`

Defined class `IniBase` with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface. It defines:

#### attributes

`__verbose` - enable/disable verbose option.  
`tool_operational` - control ATS operational functionality.  
`ini2obj` - in API for informations.  
`obj2ini` - out API for informations.  
`option_parser` - option parser for ATS.

#### methods

`__init__` - initial constructor.  
`is_tool_ok` - checking is tool operational.  
`__str__` - str dunder method for object `IniBase`.

**VERBOSE** = 'ats\_utilities::config\_io::ini'

`is_tool_ok()`

Checking is tool operational.

**Returns** boolean statusm True (yes) | False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.config\_io.json package

### Submodules

#### ats\_utilities.config\_io.json.json2object module

**Module** json2object.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Json2Object with attribute(s) and method(s). Created API for reading a configuration/information from a json file.

**class** `ats_utilities.config_io.json.json2object.Json2Object` (*configuration\_file*,  
*verbose=False*)  
Bases: `ats_utilities.config_io.base_read.BaseReadConfig`

Defined class Json2Object with attribute(s) and method(s). Created API for reading a configuration/information from a json file. It defines:

#### attributes

`__FORMAT` - format of configuration content.

`__verbose` - enable/disable verbose option.

#### methods

`__init__` - initial constructor.

`read_configuration` - read configuration from file.

`__str__` - str dunder method for object Json2Object.

`VERBOSE = 'ats_utilities::config_io::json::json2object'`

`__Json2Object__FORMAT = 'json'`

`read_configuration` (*verbose=False*)

Read configuration from a json file.

**Parameters** `verbose` (<bool>) – enable/disable verbose option.

**Returns** configuration object | None.

**Return type** <Python object(s)> | <NoneType>

**Exceptions** None

## ats\_utilities.config\_io.json.object2json module

**Module** object2json.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Object2Json with attribute(s) and method(s). Created API for writing a configuration/information to a json file.

**class** ats\_utilities.config\_io.json.object2json.**Object2Json** (*configuration\_file*,  
*verbose=False*)  
Bases: *ats\_utilities.config\_io.base\_write.BaseWriteConfig*

Defined class Object2Json with attribute(s) and method(s). Created API for writing a configuration/information to a json file. It defines:

### attributes

`__FORMAT` - format of configuration content.  
`__verbose` - enable/disable verbose option.

### methods

`__init__` - initial constructor.  
`write_configuration` - write configuration to a json file.  
`__str__` - str dunder method for object Object2Json.

`VERBOSE = 'ats_utilities::config_io::json::object2json'`

`__Object2Json__FORMAT = 'json'`

`write_configuration` (*configuration*, *verbose=False*)  
Write configuration to a json file.

### Parameters

- **verbose** (<bool>) – enable/disable verbose option.
- **configuration** – configuration object.

**Type** <Python object(s)>

**Returns** boolean status, True (success) | False.

**Return type** <bool>

**Exception** None

## Module contents

**Module** \_\_init\_\_.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied

warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class JsonBase with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface.

**class** `ats_utilities.config_io.json.JsonBase` (*informations\_file, verbose=False*)  
Bases: `object`

Defined class JsonBase with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface. It defines:

**attributes**

`__verbose` - enable/disable verbose option.  
`tool_operational` - control ATS operational functionality.  
`json2obj` - in API for informations.  
`obj2json` - out API for informations.  
`option_parser` - option parser for ATS.

**methods**

`__init__` - initial constructor.  
`is_tool_ok` - checking is tool operational.  
`__str__` - str dunder method for object JsonBase.

**VERBOSE** = `'ats_utilities::config_io::json'`

**is\_tool\_ok** ()

Checking is tool operational.

**Returns** boolean status, True (yes) | False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.config\_io.xml package

### Submodules

#### ats\_utilities.config\_io.xml.object2xml module

**Module** `object2xml.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class Object2Xml with attribute(s) and method(s). Created API for writing a configuration/information to a xml file.

```
class ats_utilities.config_io.xml.object2xml.Object2Xml (configuration_file, verbose=False)
```

Bases: *ats\_utilities.config\_io.base\_write.BaseWriteConfig*

Defined class Object2Xml with attribute(s) and method(s). Created API for writing a configuration/information to a xml file. It defines:

**attributes**

`__FORMAT` - format of configuration content.  
`__verbose` - enable/disable verbose option.

**methods**

`__init__` - initial constructor.  
`write_configuration` - write configuration to a xml file.  
`__str__` - str dunder method for object Object2Xml.

```
VERBOSE = 'ats_utilities::config_io::xml::object2xml'
```

```
_Object2Xml__FORMAT = 'xml'
```

```
write_configuration (configuration, verbose=False)
```

Write configuration to a xml file.

**Parameters**

- **configuration** – configuration object.
- **verbose** (<bool>) – enable/disable verbose option.

**Type** <BeautifulSoup>

**Returns** boolean status, True (success) | False.

**Return type** <bool>

**Exception** None

## ats\_utilities.config\_io.xml.xml2object module

**Module** xml2object.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Xml2Object with attribute(s) and method(s). Created API for reading a configuration/information from a xml file.

```
class ats_utilities.config_io.xml.xml2object.Xml2Object (configuration_file, verbose=False)
```

Bases: *ats\_utilities.config\_io.base\_read.BaseReadConfig*

Defined class Xml2Object with attribute(s) and method(s). Created API for reading a configuration/information from a xml file. It defines:

**attributes**

`__FORMAT` - format of configuration content.

`__verbose` - enable/disable verbose option.

#### methods

`__init__` - initial constructor.

`read_configuration` - read a configuration from file.

`__str__` - str dunder method for object `Xml2Object`.

```
VERBOSE = 'ats_utilities::config_io::xml::xml2object'
```

```
_Xml2Object__FORMAT = 'xml'
```

```
read_configuration(verbose=False)
```

Read a configuration from a xml file.

**Parameters** `verbose` (<bool>) – enable/disable verbose option.

**Returns** configuration object | None.

**Return type** <BeautifulSoup> | <NoneType>

**Exceptions** None

## Module contents

Module `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `XmlBase` with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface.

```
class ats_utilities.config_io.xml.XmlBase(informations_file, verbose=False)
```

Bases: object

Defined class `XmlBase` with attribute(s) and method(s). Load ATS configuration/information, setup ATS CL interface. It defines:

#### attributes

`__verbose` - enable/disable verbose option.

`tool_operational` - control ATS operational functionality.

`xml2obj` - in API for informations.

`obj2xml` - out API for informations.

`option_parser` - option parser for ATS.

#### methods

`__init__` - initial constructor.

`is_tool_ok` - checking is tool operational.

`__str__` - str dunder method for object `XmlBase`.

```
VERBOSE = 'ats_utilities::config_io::xml'
```



`is_tool_ok()`

Checking is tool operational.

**Returns** boolean status, True (yes) | False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.config\_io.yaml package

### Submodules

#### ats\_utilities.config\_io.yaml.object2yaml module

**Module** object2yaml.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <elektron.ronca@gmail.com> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Object2Yaml with attribute(s) and method(s). Created API for writing a configuration/information to a yaml file.

**class** `ats_utilities.config_io.yaml.object2yaml.Object2Yaml` (*configuration\_file*,  
*verbose=False*)  
Bases: `ats_utilities.config_io.base_write.BaseWriteConfig`

Defined class Object2Yaml with attribute(s) and method(s). Created API for writing a configuration/information to a yaml file. It defines:

#### attributes

`__FORMAT` - format of configuration content.

`__verbose` - enable/disable verbose option.

#### methods

`__init__` - initial constructor.

`write_configuration` - write configuration to a yaml file.

`__str__` - str dunder method for object Object2Yaml.

`VERBOSE = 'ats_utilities::config_io::yaml::object2yaml'`

`_Object2Yaml__FORMAT = 'yaml'`

`write_configuration` (*configuration*, *verbose=False*)

Write configuration to a yaml file.

#### Parameters

- **configuration** (<Python object(s)> | <NoneType>) – configuration object | None.

- **verbose** (<bool>) – enable/disable verbose option.

**Returns** boolean status, True (success) | False.

**Return type** <bool>

**Exception** None

## ats\_utilities.config\_io.yaml.yaml2object module

**Module** yaml2object.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Yaml2Object with attribute(s) and method(s). Created API for reading a configuration/information from a yaml file.

**class** ats\_utilities.config\_io.yaml.yaml2object.Yaml2Object (*configuration\_file*,  
*verbose=False*)

Bases: *ats\_utilities.config\_io.base\_read.BaseReadConfig*

Defined class Yaml2Object with attribute(s) and method(s). Created API for reading a configuration/information from a yaml file. It defines:

### attributes

`__FORMAT` - format of configuration content.

`__verbose` - enable/disable verbose option.

### methods

`__init__` - initial constructor.

`read_configuration` - getting a configuration from file.

`__str__` - str dunder method for object Yaml2Object.

**VERBOSE** = 'ats\_utilities::config\_io::yaml::yaml2object'

**\_\_Yaml2Object\_\_FORMAT** = 'yaml'

**read\_configuration** (*verbose=False*)

Getting a configuration from yaml file.

**Parameters** **verbose** (<bool>) – enable/disable verbose option.

**Returns** configuration object | None.

**Return type** <Python object(s)> | <NoneType>

**Exceptions** None

## Module contents

**Module** \_\_init\_\_.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied

warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `YamlBase` with attribute(s) and method(s). Load ATS information/configuration, setup ATS CL interface.

**class** `ats_utilities.config_io.yaml.YamlBase` (*informations\_file, verbose=False*)  
Bases: `object`

Defined class `YamlBase` with attribute(s) and method(s). Load ATS information/configuration, setup ATS CL interface. It defines:

#### attributes

`__verbose` - enable/disable verbose option.  
`tool_operational` - control ATS operational functionality.  
`yaml2obj` - in API for informations.  
`obj2yaml` - out API for informations.  
`option_parser` - option parser for ATS.

#### methods

`__init__` - initial constructor.  
`is_tool_ok` - checking is tool operational.  
`__str__` - str dunder method for object `YamlBase`.

**VERBOSE** = `'ats_utilities::config_io::yaml'`

**is\_tool\_ok** ()

Checking is tool operational.

**Returns** boolean status, True (yes) | False.

**Return type** <bool>

**Exceptions** None

## Submodules

### ats\_utilities.config\_io.base\_check module

**Module** `base_check.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `FileChecking` with attribute(s) and method(s). Created API for checking operations with files.

**class** `ats_utilities.config_io.base_check.FileChecking` (*verbose=False*)  
Bases: `object`

Defined class `FileChecking` with attribute(s) and method(s). Created API for checking operations with files. It defines:

### attributes

MODES - mode file operations.  
\_\_verbose - enable/disable verbose option.  
\_\_file\_path\_ok - file exist, path ok.  
\_\_file\_mode\_ok - supported file mode.  
\_\_file\_format\_ok - file format is (not) expected.

### methods

\_\_init\_\_ - initial constructor.  
check\_path - check file path.  
check\_mode - check operation mode for file.  
check\_format - check file format by extension.  
is\_file\_ok - status of file for processing.  
\_\_str\_\_ - str dunder method for FileChecking.

**MODES** = ['r', 'w', 'a', 'b', 'x', 't', '+']

**VERBOSE** = 'ats\_utilities::config\_io::base\_check'

**check\_format** (*file\_path*, *file\_format*, *verbose=False*)  
Check file format by extension.

#### Parameters

- **file\_path** (<str>) – file path.
- **file\_format** (<str>) – file format (file extension).
- **verbose** (<bool>) – enable/disable verbose option.

**Exceptions** None

**check\_mode** (*file\_mode*, *verbose=False*)  
Check operation mode for file.

#### Parameters

- **file\_mode** (<str>) – file mode ('r', 'w', 'a', 'b', 'x', 't', '+').
- **verbose** (<bool>) – enable/disable verbose option.

**Returns** True (regular file mode) | False.

**Return type** <bool>

**Exceptions** None

**check\_path** (*file\_path*, *verbose=False*)  
Check file path.

#### Parameters

- **file\_path** (<str>) – file path.
- **verbose** (<bool>) – enable/disable verbose option.

**Exceptions** None

**is\_file\_ok** ()  
Status of file for processing.

**Returns** boolean status, True (correct file) | False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.config\_io.base\_read module

**Module** base\_read.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class BaseReadConfig with attribute(s) and method(s). Created API for read operation for information/configuration files.

**class** ats\_utilities.config\_io.base\_read.**BaseReadConfig** (*verbose=False*)  
Bases: object

Defined class BaseReadConfig with attribute(s) and method(s). Created API for read operation for information/configuration files. It defines:

### attributes

`__verbose` - enable/disable verbose option.  
`__file_path` - configuration file path.

### methods

`__init__` - initial constructor.  
`file_path` - property methods for set/get operations.  
`is_not_none` - checking is file path None.  
`read_configuration` - read configuration (Abstract method).  
`__str__` - str dunder method for BaseReadConfig.

**VERBOSE** = 'ats\_utilities::config\_io::base\_read'

### file\_path

Property method for getting file path.

**Returns** configuration file path | None.

**Return type** <str> | <NoneType>

**Exception** None

### is\_not\_none()

Checking is file path None.

**Returns** boolean status, True (file path ok) | False.

**Return type** <bool>

**Exceptions** None

**read\_configuration** = <ats\_utilities.abstract.AbstractMethod object>

## ats\_utilities.config\_io.base\_write module

**Module** base\_write.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class BaseWriteConfig with attribute(s) and method(s). Created API for write operation for configuration files.

**class** ats\_utilities.config\_io.base\_write.**BaseWriteConfig** (*verbose=False*)  
Bases: object

Defined class BaseWriteConfig with attribute(s) and method(s). Created API for write operation for configuration files. It defines:

### attributes

`__verbose` - enable/disable verbose option.  
`__file_path` - configuration file path.

### methods

`__init__` - initial constructor.  
`file_path` - property methods for set/get operations.  
`is_not_none` - checking is file path None.  
`write_configuration` - write configuration (Abstract method).  
`__str__` - str dunder method for BaseWriteConfig.

**VERBOSE** = 'ats\_utilities::config\_io::base\_write'

### file\_path

Property method for getting file path.

**Returns** configuration file path | None.

**Return type** <str> | <NoneType>

**Exceptions** None

### is\_not\_none()

Checking is file path None.

**Returns** boolean status, True (file path ok) | False.

**Return type** <bool>

**Exceptions** None

**write\_configuration** = <ats\_utilities.abstract.AbstractMethod object>

## Module contents

**Module** \_\_init\_\_.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ConfigFile with attribute(s) and method(s). Created API for information/configuration context manager.

**class** `ats_utilities.config_io.ConfigFile` (*file\_path, file\_mode, file\_format, verbose=False*)  
Bases: `ats_utilities.config_io.base_check.FileChecking`

Defined class ConfigFile with attribute(s) and method(s). Created API for information/configuration context manager. It defines:

**attributes**

`__verbose` - enable/disable verbose option.  
`__file_path` - configuration file name.  
`__file_mode` - file mode.  
`__file_format` - file format.  
`__file` - file object.

**methods**

`__init__` - initial constructor.  
`__enter__` - open configuration file in mode.  
`__exit__` - close configuration file.  
`__str__` - str dunder method for ConfigFile.

**VERBOSE** = `'ats_utilities::config_io'`

### 1.1.1.5 ats\_utilities.console\_io package

#### Submodules

#### ats\_utilities.console\_io.error module

**Module** error.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSError with attribute(s) and method(s). Created error message container for console log mechanism.

**class** `ats_utilities.console_io.error.ATSError`  
Bases: `object`

Defined class ATSError with attribute(s) and method(s). Created error message container for console log mechanism. It defines:

#### attributes

`__message` - error message container.

#### methods

`__init__` - initial constructor.

`message` - property methods for set/get operations.

`is_not_none` - checking is message None or not.

`__str__` - str dunder method for ATSError.

#### `is_not_none()`

Checking is message None or not.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

#### `message`

Property method for getting message.

**Returns** formatted errors message.

**Return type** <str>

**Exceptions** None

`ats_utilities.console_io.error.error_message(error_path, *message)`  
Show error message.

#### Parameters

- **error\_path** (<str>) – error prefix message.
- **message** (<tuple>) – message parts.

**Exceptions** ATSTypeError | ATSBadCallError

## ats\_utilities.console\_io.success module

**Module** success.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSSuccess with attribute(s) and method(s). Created success message container for console log mechanism.

**class** `ats_utilities.console_io.success.ATSSuccess`

Bases: object

Defined class ATSSuccess with attribute(s) and method(s). Created success message container for console log mechanism. It defines:

#### attributes



\_\_message - success message container.

#### methods

\_\_init\_\_ - initial constructor.

message - property methods for set/get operations.

is\_not\_none - checking is message None or not.

\_\_str\_\_ - str dunder method for ATSSuccess.

#### is\_not\_none ()

Checking is message None or not.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

#### message

Property method for getting message.

**Returns** formatted success message.

**Return type** <str>

**Exceptions** None

ats\_utilities.console\_io.success.**success\_message** (*success\_path*, \**message*)

Show success message.

#### Parameters

- **success\_path** (<str>) – success prefix message.
- **message** (<tuple>) – message parts.

**Exceptions** ATSTypeError | ATSBadCallError

## ats\_utilities.console\_io.verbose module

**Module** verbose.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSVerbose with attribute(s) and method(s). Created verbose message container for console log mechanism.

**class** ats\_utilities.console\_io.verbose.**ATSVerbose**

Bases: object

Defined class ATSVerbose with attribute(s) and method(s). Created verbose message container for console log mechanism. It defines:

#### attributes

\_\_message - verbose message container.

#### methods

`__init__` - initial constructor.  
`message` - property methods for set/get operations.  
`is_not_none` - checking is message None or not.  
`__str__` - str dunder method for ATSVerbose.

#### `is_not_none()`

Checking is message None or not.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

#### `message`

Property method for getting message.

**Returns** formatted verbose message.

**Return type** <str>

`ats_utilities.console_io.verbose.verbose_message(verbose_path, verbose=False, *message)`

Show verbose message.

#### Parameters

- **verbose\_path** (<str>) – verbose prefix message.
- **verbose** (<bool>) – enable/disable verbose option.
- **message** (<tuple>) – message parts.

**Exceptions** ATSTypeError | ATSBadCallError

## ats\_utilities.console\_io.warning module

**Module** warning.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSWarning with attribute(s) and method(s). Created warning message container for console log mechanism.

**class** `ats_utilities.console_io.warning.ATSWarning`

Bases: object

Defined class ATSWarning with attribute(s) and method(s). Created warning message container for console log mechanism. It defines:

#### attributes

`__message` - warning message container.

#### methods

`__init__` - initial constructor.  
`message` - property methods for set/get operations.  
`is_not_none` - checking is message None or not.  
`__str__` - str dunder method for ATSWarning.

**is\_not\_none** ()

Checking is message None or not.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

**message**

Property method for getting message.

**Returns** formatted warning message.

**Return type** <str>

**Exceptions** None

`ats_utilities.console_io.warning.warning_message` (*warning\_path*, \**message*)

Show warning message.

**Parameters**

- **warning\_path** (<str>) – warning prefix message.
- **message** (<tuple>) – message parts.

**Exceptions** ATSTypeError | ATSBadCallError

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined package `ats_utilities.console_io`.

### 1.1.1.6 `ats_utilities.cooperative` package

#### Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General

Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class CooperativeMeta with attribute(s) and method(s). Created cooperative mechanism for multi-inheritance.

**class** `ats_utilities.cooperative.CooperativeMeta` (*name, bases, members*)

Bases: `type`

Defined class CooperativeMeta with attribute(s) and method(s). Created cooperative mechanism for multi-inheritance. It defines:

**attributes**

`combined metas` - combined metaclasses in form of dictionary.

**methods**

`__new__` - new constructor.

`__init__` - initial constructor.

`__str__` - str dunder method for CooperativeMeta.

### 1.1.1.7 `ats_utilities.exceptions` package

#### Submodules

#### `ats_utilities.exceptions.ats_attribute_error` module

**Module** `ats_attribute_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSAttributeError` with attribute(s) and method(s). Created exception for mechanism checking object attribute.

**exception** `ats_utilities.exceptions.ats_attribute_error.ATSAttributeError`

Bases: `AttributeError`

Defined class `ATSAttributeError` with attribute(s) and method(s). Created exception for mechanism checking object attribute. It defines:

**attributes**

None

**methods**

None

#### `ats_utilities.exceptions.ats_bad_call_error` module

**Module** `ats_bad_call_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSBadCallError with attribute(s) and method(s). Created exception for checking method/function argument(s) mechanism.

**exception** `ats_utilities.exceptions.ats_bad_call_error.ATSBadCallError`

Bases: Exception

Defined class ATSBadCallError with attribute(s) and method(s). Created exception for checking method/function argument(s) mechanism. It defines:

**attributes**

None

**methods**

None

### **ats\_utilities.exceptions.ats\_file\_error module**

**Module** `ats_file_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSFileError with attribute(s) and method(s). Created exception for checking file mechanism.

**exception** `ats_utilities.exceptions.ats_file_error.ATSFileError`

Bases: OSError

Defined class ATSFileError with attribute(s) and method(s). Created exception for checking file mechanism. It defines:

**attributes**

None

**methods**

None

### **ats\_utilities.exceptions.ats\_key\_error module**

**Module** `ats_key_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities

is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSKeyError` with attribute(s) and method(s). Created exception for checking key mechanism.

**exception** `ats_utilities.exceptions.ats_key_error.ATSKeyError`

Bases: `KeyError`

Defined class `ATSKeyError` with attribute(s) and method(s). Created exception for checking key mechanism. It defines:

**attributes**

None

**methods**

None

### **ats\_utilities.exceptions.ats\_lookup\_error module**

**Module** `ats_lookup_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSLookupError` with attribute(s) and method(s). Created exception for Lookup errors mechanism.

**exception** `ats_utilities.exceptions.ats_lookup_error.ATSLookupError`

Bases: `LookupError`

Defined class `ATSLookupError` with attribute(s) and method(s). Created exception for Lookup errors mechanism. It defines:

**attributes**

None

**methods**

None

### **ats\_utilities.exceptions.ats\_parameter\_error module**

**Module** `ats_parameter_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General

Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `ATSPParameterError` with attribute(s) and method(s). Created exception mechanism (raise error for not expected format).

**exception** `ats_utilities.exceptions.ats_parameter_error.ATSPParameterError`

Bases: `Exception`

Defined class `ATSPParameterError` with attribute(s) and method(s). Created exception mechanism (raise error for not expected format). It defines:

**attributes**

None

**methods**

None

### `ats_utilities.exceptions.ats_type_error` module

**Module** `ats_type_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `ATSTypeError` with attribute(s) and method(s). Created exception for check type mechanism.

**exception** `ats_utilities.exceptions.ats_type_error.ATSTypeError`

Bases: `TypeError`

Defined class `ATSTypeError` with attribute(s) and method(s). Created exception for check type mechanism. It defines:

**attributes**

None

**methods**

None

### `ats_utilities.exceptions.ats_value_error` module

**Module** `ats_value_error.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `ATSValueError` with attribute(s) and method(s). Created exception for check value mechanism.

**exception** `ats_utilities.exceptions.ats_value_error.ATSValueError`

Bases: `ValueError`

Defined class `ATSValueError` with attribute(s) and method(s). Created exception for check value mechanism. It defines:

**attributes**

None

**methods**

None

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSError` with attribute(s) and method(s). Created general exception (raise exception for wrong state/behavior).

**exception** `ats_utilities.exceptions.ATSError`

Bases: `Exception`

Defined class `ATSError` with attribute(s) and method(s). Created general exception (raise exception for wrong state/behavior). It defines:

**attributes**

None

**methods**

None

### 1.1.1.8 `ats_utilities.final` package

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSFinal` with attribute(s) and method(s). Created API for App/Tool/Script utilities, set final class.



**class** `ats_utilities.final.ATSFinal` (*name, bases, namespace*)

Bases: `type`

Defined class ATSFinal with attribute(s) and method(s). Created API for App/Tool/Script utilities, set final class. It defines:

**attributes**

`registry` - set of classes.

**methods**

`__init__` - initial constructor.

`__str__` - str dunder method for ATSFinal.

### 1.1.1.9 `ats_utilities.info` package

#### Submodules

#### `ats_utilities.info.ats_build_date` module

**Module** `ats_build_date.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSBuildDate` with attribute(s) and method(s). Created API for ATS build date in one property object.

**class** `ats_utilities.info.ats_build_date.ATSBuildDate` (*verbose=False*)

Bases: `object`

Defined class `ATSBuildDate` with attribute(s) and method(s). Created API for ATS build date in one property object. It defines:

**attributes**

`__verbose` - enable/disable verbose option.

`__build_date` - ATS build date.

**methods**

`__init__` - initial constructor.

`build_date` - property methods for set/get operations.

`is_not_none` - checking is ATS build date None.

`__str__` - str dunder method for `ATSBuildDate`.

**VERBOSE** = `'ats_utilities::info::ats_build_date'`

**build\_date**

Property method for getting ATS build date.

**Returns** ATS build date | None.

**Return type** <str> | <NoneType>

**Exceptions** None

**is\_not\_none** ()

Checking is ATS build date None.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.info.ats\_info\_ok module

**Module** ats\_info\_ok.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATInfoOk with attribute(s) and method(s). Created API for ATS info status in one property object.

**class** ats\_utilities.info.ats\_info\_ok.**ATInfoOk** (*verbose=False*)

Bases: object

Defined class ATInfoOk with attribute(s) and method(s). Created API for ATS info status in one property object. It defines:

### attributes

\_\_verbose - enable/disable verbose option.

\_\_ats\_info\_ok - ATS information status.

### methods

\_\_init\_\_ - initial constructor.

ats\_info\_ok - property methods for set/get operations.

\_\_str\_\_ - str dunder method for ATInfoOk.

**VERBOSE** = 'ats\_utilities::info::ats\_info\_ok'

**ats\_info\_ok**

Property method for getting ATS information status.

**Returns** boolean status, ATS information status.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.info.ats\_licence module

**Module** ats\_licence.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the

Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `ATSLicence` with attribute(s) and method(s). Created API for ATS licence in one property object.

**class** `ats_utilities.info.ats_licence.ATSLicence` (*verbose=False*)

Bases: `object`

Defined class `ATSLicence` with attribute(s) and method(s). Created API for ATS licence in one property object. It defines:

**attributes**

`__verbose` - enable/disable verbose option.

`__licence` - ATS licence.

**methods**

`__init__` - initial constructor.

`name` - property methods for set/get operations.

`is_not_none` - checking is ATS licence None.

`__str__` - str dunder method for `ATSLicence`.

**VERBOSE** = `'ats_utilities::info::ats_licence'`

**is\_not\_none** ()

Checking is ATS licence None.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

**licence**

Property method for getting ATS licence.

**Returns** ATS licence | None.

**Return type** <str> | <NoneType>

**Exceptions** None

## **ats\_utilities.info.ats\_name module**

**Module** `ats_name.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

**Info** Defined class `ATSName` with attribute(s) and method(s). Created API for ATS name in one property object.

**class** `ats_utilities.info.ats_name.ATSName` (*verbose=False*)

Bases: `object`

Defined class `ATSName` with attribute(s) and method(s). Created API for ATS name in one property object. It defines:

**attributes**

`__verbose` - enable/disable verbose option.

`__name` - ATS name.

**methods**

`__init__` - initial constructor.

`name` - property methods for set/get operations.

`is_not_none` - checking is ATS name None.

`__str__` - str dunder method for `ATSName`.

**VERBOSE** = `'ats_utilities::info::ats_name'`

**is\_not\_none** ()

Checking is ATS name None.

**Returns** boolean status, True | False.

**Return type** `<bool>`

**Exceptions** None

**name**

Property method for getting ATS name.

**Returns** ATS name | None.

**Return type** `<str>` | `<NoneType>`

**Exceptions** None

## ats\_utilities.info.ats\_version module

**Module** `ats_version.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSVersion` with attribute(s) and method(s). Created API for ATS version in one property object.

**class** `ats_utilities.info.ats_version.ATSVersion` (*verbose=False*)

Bases: `object`

Defined class `ATSVersion` with attribute(s) and method(s). Created API for ATS version in one property object. It defines:

**attributes**

`__verbose` - enable/disable verbose option.

`__version` - ATS version.

### methods

`__init__` - initial constructor.  
`version` - property methods for set/get operations.  
`is_not_none` - checking is ATS version None.  
`__str__` - str dunder method for ATSVersion.

**VERBOSE** = 'ats\_utilities::info::ats\_version'

**is\_not\_none** ()

Checking is ATS version None.

**Returns** boolean status, True | False.

**Return type** <bool>

**Exceptions** None

### version

Property method for getting ATS version.

**Returns** ATS version | None.

**Return type** <str> | <NoneType>

**Exceptions** None

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSInfo` with attribute(s) and method(s). Created API for ATS informations in one container object.

**class** `ats_utilities.info.ATSInfo` (*info, verbose=False*)

Bases: `ats_utilities.info.ats_name.ATSName`, `ats_utilities.info.ats_version.ATSVersion`, `ats_utilities.info.ats_licence.ATSLicence`, `ats_utilities.info.ats_build_date.ATSBuildDate`, `ats_utilities.info.ats_info_ok.ATSInfoOk`

Defined class `ATSInfo` with attribute(s) and method(s). Created API for ATS informations in one container object. It defines:

### attributes

`__metaclass__` - setting cooperative metaclasses.  
`ATS_NAME` - ATS name key.  
`ATS_VERSION` - ATS version key.  
`ATS_LICENCE` - ATS licence key.  
`ATS_BUILD_DATE` - ATS build date key.  
`ATS_BASE_INFO` - ATS base information dict.  
`__verbose` - enable/disable verbose option.

### methods

`__init__` - initial constructor.  
`show_base_info` - show ATS informations.  
`is_correct` - check information structure.  
`__str__` - str dunder method for ATSInfo.

```
ATS_BASE_INFO = {1: 'ats_name', 2: 'ats_version', 3: 'ats_licence', 4: 'ats_build_
```

```
ATS_BUILD_DATE = 'ats_build_date'
```

```
ATS_LICENCE = 'ats_licence'
```

```
ATS_NAME = 'ats_name'
```

```
ATS_VERSION = 'ats_version'
```

```
VERBOSE = 'ats_utilities::info'
```

`is_correct` (*informations*, *verbose=False*)  
Check information structure.

### Parameters

- **informations** (*<dict>*) – ATS base informations.
- **verbose** (*<bool>*) – enable/disable verbose option.

**Exceptions** `ATSTypeError` | `ATSBadCallError`

`show_base_info` ()  
Show ATS informations.

**Exceptions** `None`

## 1.1.1.10 ats\_utilities.logging package

### Submodules

#### ats\_utilities.logging.ats\_logger\_file module

**Module** `ats_logger_file.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSLoggerFile` with attribute(s) and method(s). Created API for ATS logger file path in one property object.

**class** `ats_utilities.logging.ats_logger_file.ATSLoggerFile` (*verbose=False*)  
Bases: `object`

Defined class `ATSLoggerFile` with attribute(s) and method(s). Created API for ATS logger file path in one property object. It defines:

### attributes

`__verbose` - enable/disable verbose option.  
`__logger_file` - log file path.

#### methods

`__init__` - initial constructor.  
`logger_file` - property methods for set/get operations.  
`__str__` - str dunder method for `ATSLoggerFile`.

**VERBOSE** = `'ats_utilities::logging::ats_logger_file'`

#### `logger_file`

Property method for getting log file path.

**Returns** log file path | None.

**Return type** <str> | <NoneType>

**Exceptions** None

### ats\_utilities.logging.ats\_logger\_name module

**Module** `ats_logger_name.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSLoggerName` with attribute(s) and method(s). Created API for ATS logger name in one property object.

**class** `ats_utilities.logging.ats_logger_name.ATSLoggerName` (*verbose=False*)  
Bases: `object`

Defined class `ATSLoggerName` with attribute(s) and method(s). Created API for ATS logger name in one property object. It defines:

#### attributes

`__verbose` - enable/disable verbose option.  
`__logger_name` - logger name.

#### methods

`__init__` - initial constructor.  
`logger_name` - property methods for set/get operations.  
`__str__` - str dunder method for `ATSLoggerName`.

**VERBOSE** = `'ats_utilities::logging::ats_logger_name'`

#### `logger_name`

Property method for getting logger name.

**Returns** logger name | None.

**Return type** <str> | <NoneType>

**Exceptions** None

## ats\_utilities.logging.ats\_logger\_status module

**Module** `ats_logger_status.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSLoggerStatus` with attribute(s) and method(s). Created API for ATS logger status in one property object.

**class** `ats_utilities.logging.ats_logger_status.ATSLoggerStatus` (*verbose=False*)  
Bases: `object`

Defined class `ATSLoggerStatus` with attribute(s) and method(s). Created API for ATS logger status in one property object. It defines:

### attributes

`__verbose` - enable/disable verbose option.  
`__log_status` - logger status (enabled/disabled).

### methods

`__init__` - initial constructor.  
`logger_status` - property methods for set/get operations.  
`__str__` - str dunder method for `ATSLoggerStatus`.

**VERBOSE** = `'ats_utilities::logging::ats_logger_status'`

### `logger_status`

Property method for getting logger status.

**Returns** True (logger operative) | False.

**Return type** <bool>

**Exceptions** None

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> `ats_utilities` is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. `ats_utilities` is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `ATSLogger` with attribute(s) and method(s). Created API for ATS logging mechanism.



**class** `ats_utilities.logging.ATSLogger` (*ats\_name, ats\_log\_file, verbose=False*)

Bases: `ats_utilities.logging.ats_logger_name.ATSLoggerName`, `ats_utilities.logging.ats_logger_file.ATSLoggerFile`, `ats_utilities.logging.ats_logger_status.ATSLoggerStatus`

Defined class `ATSLogger` with attribute(s) and method(s). Created API for ATS logging mechanism. It defines:

**attributes**

`LOG_MSG_FORMAT` - log message format.  
`LOG_DATE_FORMAT` - log date format.  
`ATS_DEBUG` - debug log level.  
`ATS_WARNING` - warning log level.  
`ATS_CRITICAL` - critical log level.  
`ATS_ERROR` - error log level.  
`ATS_INFO` - info log level.  
`__verbose` - enable/disable verbose option.

**methods**

`__init__` - initial constructor.  
`write_log` - write message to log file.  
`__str__` - str dunder method for `ATSLogger`.

`ATS_CRITICAL = 50`

`ATS_DEBUG = 10`

`ATS_ERROR = 40`

`ATS_INFO = 20`

`ATS_WARNING = 30`

`LOG_DATE_FORMAT = '%m/%d/%Y %I:%M:%S %p'`

`LOG_MSG_FORMAT = '%(asctime)s - %(levelname)s - %(message)s'`

`VERBOSE = 'ats_utilities::logging'`

`write_log` (*message, ctrl, verbose=False*)

Write message to log file.

**Parameters**

- **message** (*<str>*) – log message.
- **ctrl** (*<int>*) – control flag (debug, warning, critical, errors, info).
- **verbose** (*<bool>*) – enable/disable verbose option.

**Returns** boolean status, True (success) | False.

**Return type** `<bool>`

**Exceptions** `ATSTypeError` | `ATSBadCallError`

### 1.1.1.11 `ats_utilities.option` package

#### Module contents

Module `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSOptionParser with attribute(s) and method(s). Created option parser and argument processor.

**class** `ats_utilities.option.ATSOptionParser` (*version, epilog, description, verbose=False*)

Bases: `object`

Defined class ATSOptionParser with attribute(s) and method(s). Created option parser and argument processor.

It defines:

**attributes**

`__verbose` - enable/disable verbose option.

`__opt_parser` - options parser.

**methods**

`__init__` - initial constructor.

`add_operation` - add option to ATS.

`parse_args` - process arguments from start.

`__str__` - str dunder method for ATSOptionParser.

**VERBOSE** = `'ats_utilities::option'`

**add\_operation** (*\*args, \*\*kwargs*)

Add option to ATS.

**Parameters**

- **args** (*<list>*) – list of arguments (objects).
- **kwargs** (*<dict>*) – arguments in shape of dictionary.

**Exceptions** None

**parse\_args** (*arguments, verbose=False*)

Process arguments from start.

**Parameters**

- **arguments** (*<list>*) – arguments.
- **verbose** (*<bool>*) – enable/disable verbose option.

**Returns** namespace object.

**Return type** `<argparse.Namespace>`

**Exceptions** None

### 1.1.1.12 `ats_utilities.register` package

#### Module contents

Module `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ATSRegister with attribute(s) and method(s). Created API for App/Tool/Script utilities, auto-register class.

**class** `ats_utilities.register.ATSRegister` (*name, bases, namespace*)  
Bases: `type`

Defined class ATSRegister with attribute(s) and method(s). Created API for App/Tool/Script utilities, auto-register class. It defines:

**attributes**

registry - set of classes.

**methods**

`__init__` - initial constructor.

`__iter__` - getting iterator of set.

`__str__` - str dunder method for ATSRegister.

### 1.1.1.13 ats\_utilities.singleton package

#### Submodules

#### ats\_utilities.singleton.base module

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class Singleton with attribute(s) and method(s). Created API for auto-register singleton object.

**class** `ats_utilities.singleton.base.Singleton`  
Bases: `object`

Defined class Singleton with attribute(s) and method(s). Created API for auto-register singleton object. It defines:

**attributes**

`__INSTANCES` - class dictionary for collecting instances.

**methods**

`__new__` - set class instance.

`__str__` - str dunder method for Singleton.

```
_Singleton__INSTANCE = None
```

## ats\_utilities.singleton.functional module

**Module** functional.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined decorator FunctionalSingleton without attribute(s). Created API for auto-register singleton object.

```
ats_utilities.singleton.functional.FunctionalSingleton (class_)
    Checking instance and setup instance.
```

**Parameters** `class (<Python Object>)` – singleton class.

**Returns** python object instance.

**Return type** <Python Object>

**Exceptions** None

## ats\_utilities.singleton.meta module

**Module** meta.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class SingletonMeta with attribute(s) and method(s). Created API for auto-register singleton object.

```
class ats_utilities.singleton.meta.SingletonMeta
    Bases: type
```

Defined class SingletonMeta with attribute(s) and method(s). Created API for auto-register singleton object. It defines:

### attributes

`__INSTANCES` - class dictionary for collecting instances.

### methods

`__call__` - check instance and setup instance.

`__str__` - str dunder method for SingletonMeta.

```
_SingletonMeta__INSTANCES = {}
```

## ats\_utilities.singleton.meta\_thread module

**Module** meta\_thread.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ThreadSafeSingletonMeta with attribute(s) and method(s). Created API for auto-register singleton object.

**class** ats\_utilities.singleton.meta\_thread.ThreadSafeSingletonMeta

Bases: type

Defined class ThreadSafeSingletonMeta with attribute(s) and method(s). Created API for auto-register singleton object. It defines:

### attributes

\_\_INSTANCES - class dictionary for collecting instances.

\_\_LOCK - thread lock object.

### methods

\_\_call\_\_ - check instance and setup instance.

\_\_str\_\_ - str dunder method for ThreadSafeSingletonMeta.

```
_ThreadSafeSingletonMeta__INSTANCES = {}
```

```
_ThreadSafeSingletonMeta__LOCK = <unlocked _thread.lock object>
```

## Module contents

**Module** \_\_init\_\_.py

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined package ats\_utilities.singleton.

### 1.1.1.14 ats\_utilities.splash package

#### Submodules

## ats\_utilities.splash.ext\_infrastructure module

**Module** ext\_infrastructure.py

**Copyright** Copyright (C) 2021 Vladimir Roncevic <elektron.ronca@gmail.com> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ExtInfrastructure with attribute(s) and method(s). Created API for processing hyperlinks for splash.

**class** ats\_utilities.splash.ext\_infrastructure.**ExtInfrastructure** (*property*, *verbose=False*)

Bases: object

Defined class ExtInfrastructure with attribute(s) and method(s). Created API for processing hyperlinks for splash. It defines:

**attributes**

\_\_verbose - enable/disable verbose option.  
\_\_property - splash property in dict format.

**methods**

\_\_init\_\_ - initial constructor.  
get\_info\_text - pre-processed info text.  
get\_issue\_text - pre-processed issue text.  
get\_author\_text - pre-processed author text.  
\_\_str\_\_ - dunder method for ExtInfrastructure.

**VERBOSE = 'ats\_utilities::splash::ext\_infrastructure'**

**get\_author\_text** ()

Pre-processed author text for splash.

**Returns** hyperlink with author info.

**Return type** <str>

**Exceptions** None

**get\_info\_text** ()

Pre-processed info text for splash.

**Returns** hyperlink with info text.

**Return type** <str>

**Exceptions** None

**get\_issue\_text** ()

Pre-processed issue text for splash.

**Returns** hyperlink with issue info.

**Return type** <str>

**Exceptions** None

## ats\_utilities.splash.github\_infrastructure module

**Module** github\_infrastructure.py

**Copyright** Copyright (C) 2021 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class GitHubInfrastructure with attribute(s) and method(s). Created API for processing hyperlinks for splash.

**class** ats\_utilities.splash.github\_infrastructure.**GitHubInfrastructure** (*property*,  
*ver-*  
*bose=False*)

Bases: object

Defined class GitHubInfrastructure with attribute(s) and method(s). Created API for processing hyperlinks for splash. It defines:

**attributes**

\_\_verbose - enable/disable verbose option.

\_\_property - splash property in dict format.

**methods**

\_\_init\_\_ - initial constructor.

get\_info\_text - pre-processed info text.

get\_issue\_text - pre-processed issue text.

get\_author\_text - pre-processed author text.

\_\_str\_\_ - dunder method for GitHubInfrastructure.

**VERBOSE = 'ats\_utilities::splash::github\_infrastructure'**

**get\_author\_text** ()

Pre-processed author text for splash.

**Returns** hyperlink with author info.

**Return type** <str>

**Exceptions** None

**get\_info\_text** ()

Pre-processed info text for splash.

**Returns** hyperlink with info text.

**Return type** <str>

**Exceptions** None

**get\_issue\_text** ()

Pre-processed issue text for splash.

**Returns** hyperlink with issue info.

**Return type** <str>

**Exceptions** None

## ats\_utilities.splash.progress\_bar module

**Module** progress\_bar.py

**Copyright** Copyright (C) 2021 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class ProgressBar with attribute(s) and method(s). Load a progressbar as part of splashscreen.

**class** ats\_utilities.splash.progress\_bar.ProgressBar (end, start=0)  
Bases: object

Defined class ProgressBar with attribute(s) and method(s). Load a progressbar as part of splashscreen. It defines:

### attributes

DEFAULT\_BAR\_LENGTH - length of progressbar.  
DEFAULT\_CHAR\_ON - loaded progress element.  
DEFAULT\_CHAR\_OFF - unloaded progress element.  
start - start of level.  
end - end of level.  
\_bar\_length - progress length.  
\_level - progress level.  
\_plotted - plotted progress.  
\_level\_chars - level progress chars.

### methods

\_\_init\_\_ - initial constructor.  
set\_level - set level of progress.  
plot\_progress - plot progress.  
set\_and\_plot - set and plot progress.  
\_\_del\_\_ - dunder method for ProgressBar.

**DEFAULT\_BAR\_LENGTH = 60**

**DEFAULT\_CHAR\_OFF = ' '**

**DEFAULT\_CHAR\_ON = ''**

**plot\_progress** (columns)  
Plot progress.

**Parameters** columns (<int>) – columns for open console session.

**Exceptions** None

**set\_and\_plot** (level, columns)  
Set and plot progress.

### Parameters

- **level** (<int>) – level of progress.
- **columns** (<int>) – columns for open console session.



**Exceptions** None

**set\_level** (*level*)

Set level.

**Parameters** **level** (<*int*>) – level of progress.

**Exceptions** None

## ats\_utilities.splash.splash\_property module

**Module** splash\_property.py

**Copyright** Copyright (C) 2021 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class SplashProperty with attribute(s) and method(s). Created API for checking splash property.

**class** ats\_utilities.splash.splash\_property.**SplashProperty** (*property*, *verbose=False*)

Bases: object

Defined class SplashProperty with attribute(s) and method(s). Created API for checking splash property. It defines:

### attributes

`__EXPECTED_PROPERTY_KEYS` - expected property names.

`__verbose` - enable/disable verbose option.

`__property` - splash property in dict format.

### methods

`__init__` - initial constructor.

`validation` - validation of splash property.

`__str__` - dunder method for SplashProperty.

`VERBOSE = 'ats_utilities::splash::splash_property'`

`__SplashProperty__EXPECTED_PROPERTY_KEYS = ['ats_organization', 'ats_repository', 'ats_n`

**validate** (*verbose=False*)

Validation of splash property.

**Parameters** **verbose** (<*bool*>) – enable/disable verbose option.

**Returns** Boolean status True - property ok else False.

**Return type** <bool>

**Exceptions** None

## ats\_utilities.splash.terminal\_properties module

**Module** terminal\_properties.py

**Copyright** Copyright (C) 2021 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class TerminalProperties with attribute(s) and method(s). Defined API for getting terminal properties.

**class** ats\_utilities.splash.terminal\_properties.**TerminalProperties** (*verbose=False*)  
Bases: object

Defined class TerminalProperties with attribute(s) and method(s). Defined API for getting terminal properties. It defines:

**attributes**

\_\_verbose - enable/disable verbose option.  
\_\_window\_size - terminal window size.

**methods**

\_\_init\_\_ - initial constructor.  
\_\_ioctl\_get\_window\_size - size for descriptor.  
\_\_ioctl\_for\_all\_descriptors - size for all descriptors.  
size - getting size of terminal window.  
\_\_str\_\_ - dunder method for TerminalProperties.

**VERBOSE = 'ats\_utilities::splash::terminal\_properties'**

**\_TerminalProperties\_\_ioctl\_for\_all\_descriptors** (*verbose=False*)  
Check window size for all descriptors.

**Parameters** **verbose** (<bool>) – enable/disable verbose option.

**Returns** status true window size checked else false.

**Return type** <bool>

**Exceptions** None

**\_TerminalProperties\_\_ioctl\_get\_window\_size** (*file\_descriptor, verbose=False*)  
Check window size for descriptor.

**Parameters**

- **file\_descriptor** (<int>) – file descriptor.
- **verbose** (<bool>) – enable/disable verbose option.

**Returns** window size of terminal.

**Return type** <tuple>

**Exceptions** ATSTypeError | ATSBadCallError

**size** (*verbose=False*)  
Center console line.

**Parameters** **verbose** (<bool>) – enable/disable verbose option.

**Returns** window size | None.

**Return type** <tuple> | <NoneType>

**Exceptions** None

## Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2021 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class `Splash` with attribute(s) and method(s). Load a splash screen info and add hyperlinks.

**class** `ats_utilities.splash.Splash` (*ats\_splash\_property*, *verbose=False*)  
Bases: `ats_utilities.splash.splash_property.SplashProperty`

Defined class `Splash` with attribute(s) and method(s). Load a splash screen info and add hyperlinks. It defines:

### attributes

`__verbose` - enable/disable verbose option.

### methods

`__init__` - initial constructor.

`center` - center console line.

`__str__` - dunder method for `Splash`.

**VERBOSE** = `'ats_utilities::splash'`

**center** (*columns*, *additional\_shifter*, *text*, *verbose=False*)  
Center console line.

### Parameters

- **columns** (<int>) – columns for open console session.
- **additional\_shifter** (<int>) – additional shifters.
- **text** (<str>) – text for console session.
- **verbose** (<bool>) – enable/disable verbose option.

**Exceptions** `ATSTypeError` | `ATSBadCallError`

## 1.1.2 Module contents

**Module** `__init__.py`

**Copyright** Copyright (C) 2017 Vladimir Roncevic <[elektron.ronca@gmail.com](mailto:elektron.ronca@gmail.com)> ats\_utilities is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. ats\_utilities is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Info** Defined class VerboseRoot with attribute(s) and method(s). Created API for setup verbose class path from metaclass.

**class** `ats_utilities.VerboseRoot`

Bases: `type`

Defined class VerboseRoot with attribute(s) and method(s). Created API for setup verbose class path from metaclass. It defines:

**attributes**

`ROOT_PACKAGE_NAME` - root package name of framework.

**methods**

`__new__` - new constructor.

`__str__` - str dunder method for VerboseRoot.

`ROOT_PACKAGE_NAME = 'ATS_UTILITIES'`

## CHAPTER 2

### Installation



```
vroncevic@debtux
-----
OS: Debian GNU/Linux 11 (bullseye) x86_64
Host: 20F5S80B00 ThinkPad X260
Kernel: 5.10.0-8-amd64
Uptime: 34 mins
Packages: 3372 (dpkg)
Shell: bash 5.1.8
Resolution: 1366x768
DE: GNOME 3.38.4
WM: Mutter
WM Theme: Adwaita
Theme: Adwaita-dark [GTK2/3]
Icons: Adwaita [GTK2/3]
Terminal: terminator
CPU: Intel i5-6200U (4) @ 2.800GHz
GPU: Intel Skylake GT2 [HD Graphics 520]
Memory: 2907MiB / 7390MiB

  _met$$$$$gg.
 ,g$$$$$$$$$$$$$$P.
 ,g$$P"         ""Y$$".
 '$$P'          '$$.
 '$$P' ,ggs.    '$$b:
 d$$'  ,P"     '$$.
 $$$   d$'     '$$P
 $$:    $$     ,d$$'
 $$;    Y$b._  ,d$P'
 Y$$    "Y$$$$P"
 `$$b
 `Y$$
 `Y$$
  $$b.
  Y$$b.
   "Y$b._
    ""
```

Navigate to release page download and extract release archive.

To install **ats\_utilities** run

```
tar xvzf ats_utilities-x.y.z.tar.gz
cd ats_utilities-x.y.z
# python2
```

(continues on next page)

(continued from previous page)

```
wget https://bootstrap.pypa.io/pip/2.7/get-pip.py
python2 get-pip.py
python2 -m pip install --upgrade setuptools
python2 -m pip install --upgrade pip
python2 -m pip install --upgrade build
pip2 install -r requirements.txt
python2 -m build --no-isolation --wheel
pip2 install dist/ats_utilities-x.y.z-py2-none-any.whl
rm -f get-pip.py
# python3
wget https://bootstrap.pypa.io/get-pip.py
python3 get-pip.py
python3 -m pip install --upgrade setuptools
python3 -m pip install --upgrade pip
python3 -m pip install --upgrade build
pip3 install -r requirements.txt
python3 -m build --no-isolation --wheel
pip3 install dist/ats_utilities-x.y.z-py3-none-any.whl
rm -f get-pip.py
```

Or type the following

```
tar xvzf ats_utilities-x.y.z.tar.gz
cd ats_utilities-x.y.z/
# python2
wget https://bootstrap.pypa.io/pip/2.7/get-pip.py
python2 get-pip.py
python2 -m pip install --upgrade setuptools
python2 -m pip install --upgrade pip
python2 -m pip install --upgrade build
pip2 install -r requirements.txt
python2 setup.py install_lib
python2 setup.py install_egg_info
# python3
wget https://bootstrap.pypa.io/get-pip.py
python3 get-pip.py
python3 -m pip install --upgrade setuptools
python3 -m pip install --upgrade pip
python3 -m pip install --upgrade build
pip3 install -r requirements.txt
python3 setup.py install_lib
python3 setup.py install_egg_info
```

You can use Docker to create image/container, or You can use pip to install

```
# python2
pip2 install ats_utilities
# python3
pip3 install ats_utilities
```

## CHAPTER 3

---

### Dependencies

---

**ats\_utilities** requires next modules and libraries

- six - Python 2 and 3 compatibility library
- yaml - YAML parser and emitter for Python
- bs4 - Screen-scraping library
- configparser - Configuration parser library
- colorama - Cross-platform colored terminal text
- pathlib - Object-oriented filesystem paths





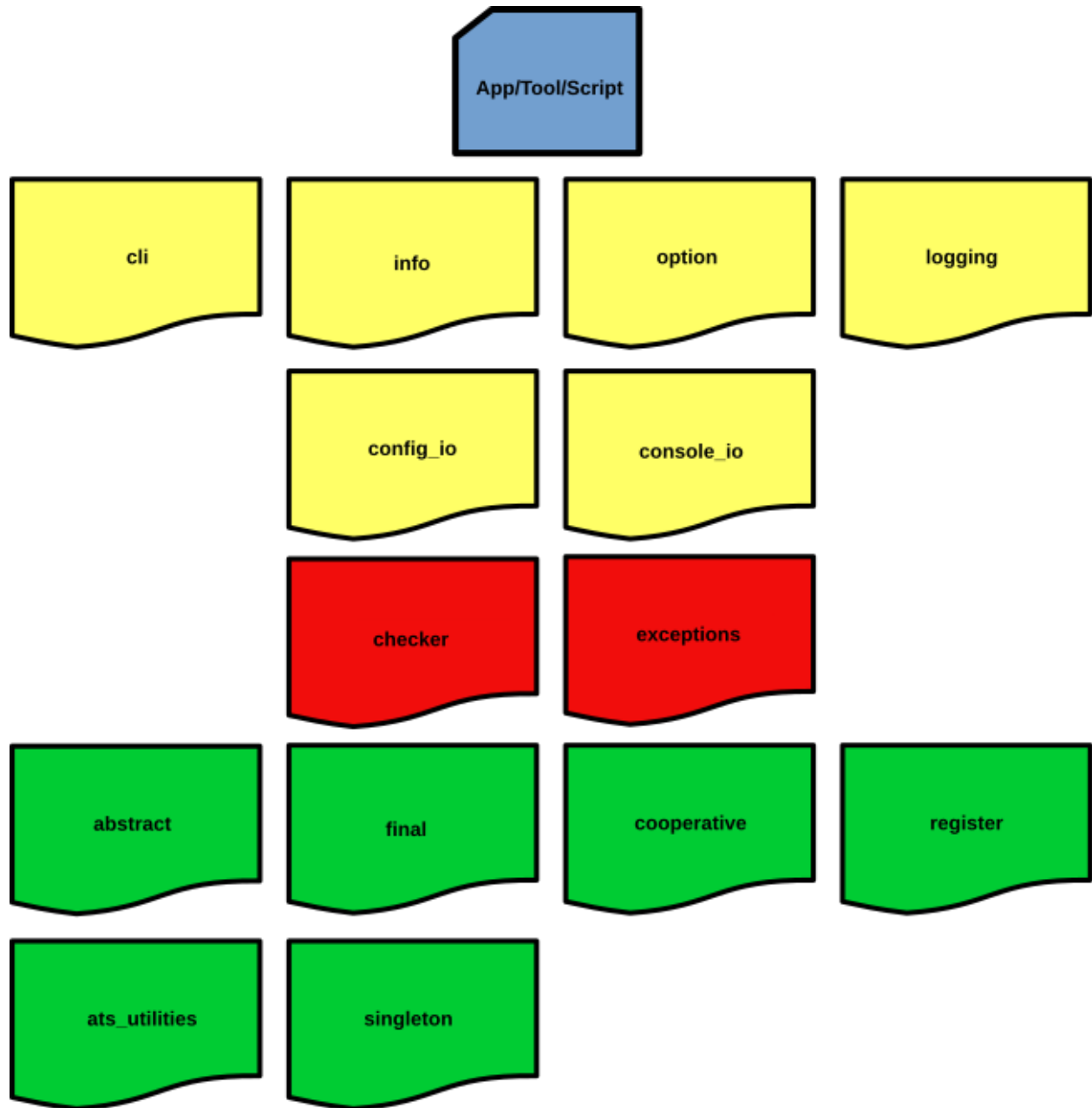
## CHAPTER 4

---

### Framework structure

---

**ats\_utilities** is based on OOP.



Framework structure

```
ats_utilities/  
├── abstract/  
│   └── __init__.py  
├── checker/  
│   └── __init__.py  
├── cli/  
│   ├── cfg_cli.py  
│   ├── ini_cli.py  
│   ├── __init__.py  
│   ├── json_cli.py  
│   ├── xml_cli.py  
│   └── yaml_cli.py
```

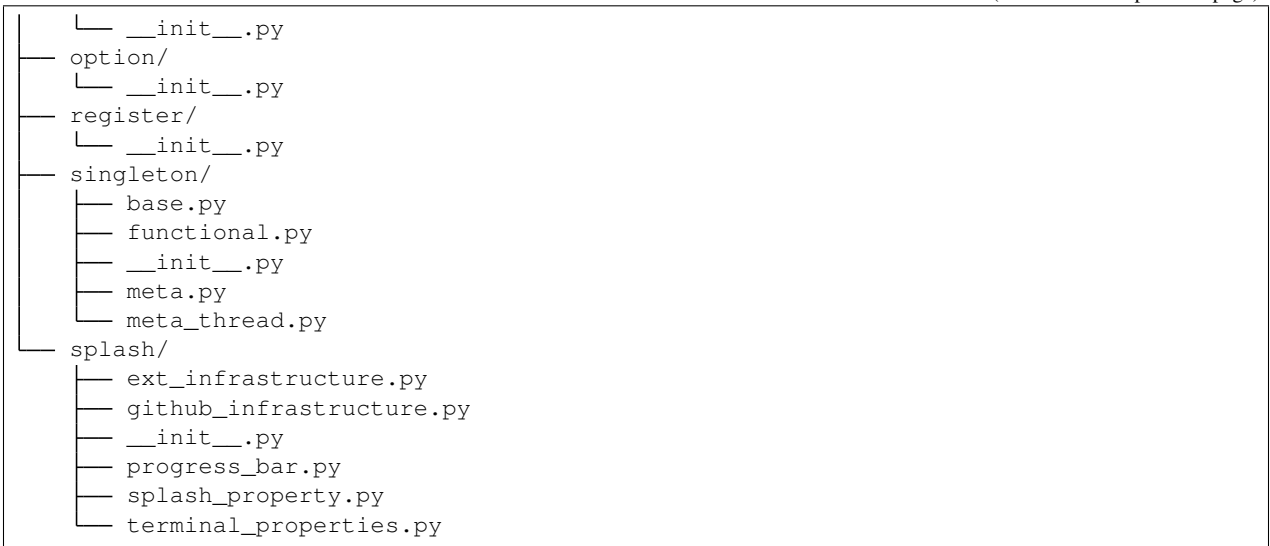
(continues on next page)

(continued from previous page)

```
— config_io/
  — base_check.py
  — base_read.py
  — base_write.py
  — cfg/
    — cfg2object.py
    — __init__.py
    — object2cfg.py
  — ini/
    — ini2object.py
    — __init__.py
    — object2ini.py
  — __init__.py
  — json/
    — __init__.py
    — json2object.py
    — object2json.py
  — xml/
    — __init__.py
    — object2xml.py
    — xml2object.py
  — yaml/
    — __init__.py
    — object2yaml.py
    — yaml2object.py
— console_io/
  — error.py
  — __init__.py
  — success.py
  — verbose.py
  — warning.py
— cooperative/
  — __init__.py
— exceptions/
  — ats_attribute_error.py
  — ats_bad_call_error.py
  — ats_file_error.py
  — ats_key_error.py
  — ats_lookup_error.py
  — ats_parameter_error.py
  — ats_type_error.py
  — ats_value_error.py
  — __init__.py
— final/
  — __init__.py
— info/
  — ats_build_date.py
  — ats_info_ok.py
  — ats_licence.py
  — ats_name.py
  — ats_version.py
  — __init__.py
— __init__.py
— logging/
  — ats_logger_file.py
  — ats_logger_name.py
  — ats_logger_status.py
```

(continues on next page)

(continued from previous page)



---

## Copyright and licence

---

Copyright (C) 2017 by [vrontcevic.github.io/ats\\_utilities](https://vrontcevic.github.io/ats_utilities)

**ats\_utilities** is free software; you can redistribute it and/or modify it under the same terms as Python itself, either Python version 2.x/3.x or, at your option, any later version of Python 3 you may have available.

Lets help and support PSF.





## CHAPTER 6

---

### Indices and tables

---

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





### a

ats\_utilities, 55  
ats\_utilities.abstract, 3  
ats\_utilities.checker, 4  
ats\_utilities.cli, 10  
ats\_utilities.cli.cfg\_cli, 6  
ats\_utilities.cli.ini\_cli, 7  
ats\_utilities.cli.json\_cli, 8  
ats\_utilities.cli.xml\_cli, 8  
ats\_utilities.cli.yaml\_cli, 9  
ats\_utilities.config\_io, 26  
ats\_utilities.config\_io.base\_check, 23  
ats\_utilities.config\_io.base\_read, 25  
ats\_utilities.config\_io.base\_write, 26  
ats\_utilities.config\_io.cfg, 12  
ats\_utilities.config\_io.cfg.cfg2object, 11  
ats\_utilities.config\_io.cfg.object2cfg, 12  
ats\_utilities.config\_io.ini, 15  
ats\_utilities.config\_io.ini.ini2object, 13  
ats\_utilities.config\_io.ini.object2ini, 14  
ats\_utilities.config\_io.json, 17  
ats\_utilities.config\_io.json.json2object, 16  
ats\_utilities.config\_io.json.object2json, 17  
ats\_utilities.config\_io.xml, 20  
ats\_utilities.config\_io.xml.object2xml, 18  
ats\_utilities.config\_io.xml.xml2object, 19  
ats\_utilities.config\_io.yaml, 22  
ats\_utilities.config\_io.yaml.object2yaml, 21  
ats\_utilities.config\_io.yaml.yaml2object, 22  
ats\_utilities.console\_io, 31  
ats\_utilities.console\_io.error, 27  
ats\_utilities.console\_io.success, 28  
ats\_utilities.console\_io.verbose, 29  
ats\_utilities.console\_io.warning, 30  
ats\_utilities.cooperative, 31  
ats\_utilities.exceptions, 36  
ats\_utilities.exceptions.ats\_attribute\_error, 32  
ats\_utilities.exceptions.ats\_bad\_call\_error, 32  
ats\_utilities.exceptions.ats\_file\_error, 33  
ats\_utilities.exceptions.ats\_key\_error, 33  
ats\_utilities.exceptions.ats\_lookup\_error, 34  
ats\_utilities.exceptions.ats\_parameter\_error, 34  
ats\_utilities.exceptions.ats\_type\_error, 35  
ats\_utilities.exceptions.ats\_value\_error, 35  
ats\_utilities.final, 36  
ats\_utilities.info, 41  
ats\_utilities.info.ats\_build\_date, 37  
ats\_utilities.info.ats\_info\_ok, 38  
ats\_utilities.info.ats\_licence, 38  
ats\_utilities.info.ats\_name, 39  
ats\_utilities.info.ats\_version, 40  
ats\_utilities.logging, 44  
ats\_utilities.logging.ats\_logger\_file, 42  
ats\_utilities.logging.ats\_logger\_name, 43  
ats\_utilities.logging.ats\_logger\_status, 44  
ats\_utilities.option, 45  
ats\_utilities.register, 46  
ats\_utilities.singleton, 49

ats\_utilities.singleton.base, [47](#)  
ats\_utilities.singleton.functional, [48](#)  
ats\_utilities.singleton.meta, [48](#)  
ats\_utilities.singleton.meta\_thread, [49](#)  
ats\_utilities.splash, [55](#)  
ats\_utilities.splash.ext\_infrastructure,  
    [49](#)  
ats\_utilities.splash.github\_infrastructure,  
    [50](#)  
ats\_utilities.splash.progress\_bar, [52](#)  
ats\_utilities.splash.splash\_property,  
    [53](#)  
ats\_utilities.splash.terminal\_properties,  
    [53](#)

## Symbols

- `_Cfg2Object__FORMAT`  
(*ats\_utilities.config\_io.cfg.cfg2object.Cfg2Object* attribute), 11
  - `_Cfg2Object__REGEX_MATCH_LINE`  
(*ats\_utilities.config\_io.cfg.cfg2object.Cfg2Object* attribute), 11
  - `_Ini2Object__FORMAT`  
(*ats\_utilities.config\_io.ini.ini2object.Ini2Object* attribute), 14
  - `_Json2Object__FORMAT`  
(*ats\_utilities.config\_io.json.json2object.Json2Object* attribute), 16
  - `_Object2Cfg__FORMAT`  
(*ats\_utilities.config\_io.cfg.object2cfg.Object2Cfg* attribute), 12
  - `_Object2Ini__FORMAT`  
(*ats\_utilities.config\_io.ini.object2ini.Object2Ini* attribute), 15
  - `_Object2Json__FORMAT`  
(*ats\_utilities.config\_io.json.object2json.Object2Json* attribute), 17
  - `_Object2Xml__FORMAT`  
(*ats\_utilities.config\_io.xml.object2xml.Object2Xml* attribute), 19
  - `_Object2Yaml__FORMAT`  
(*ats\_utilities.config\_io.yaml.object2yaml.Object2Yaml* attribute), 21
  - `_SingletonMeta__INSTANCES`  
(*ats\_utilities.singleton.meta.SingletonMeta* attribute), 48
  - `_Singleton__INSTANCE`  
(*ats\_utilities.singleton.base.Singleton* attribute), 47
  - `_SplashProperty__EXPECTED_PROPERTY_KEYS`  
(*ats\_utilities.splash.splash\_property.SplashProperty* attribute), 53
  - `_TerminalProperties__ioctl_for_all_descriptors`  
(*ats\_utilities.splash.terminal\_properties.TerminalProperties* method), 54
  - `_TerminalProperties__ioctl_get_window_size()`  
(*ats\_utilities.splash.terminal\_properties.TerminalProperties* method), 54
  - `_ThreadSafeSingletonMeta__INSTANCES`  
(*ats\_utilities.singleton.meta\_thread.ThreadSafeSingletonMeta* attribute), 49
  - `_ThreadSafeSingletonMeta__LOCK`  
(*ats\_utilities.singleton.meta\_thread.ThreadSafeSingletonMeta* attribute), 49
  - `_Xml2Object__FORMAT`  
(*ats\_utilities.config\_io.xml.xml2object.Xml2Object* attribute), 20
  - `_Yaml2Object__FORMAT`  
(*ats\_utilities.config\_io.yaml.yaml2object.Yaml2Object* attribute), 22
- ## A
- `AbstractMethod` (class in *ats\_utilities.abstract*), 3
  - `add_new_option()` (*ats\_utilities.cli.cfg\_cli.CfgCLI* method), 6
  - `add_new_option()` (*ats\_utilities.cli.ini\_cli.IniCLI* method), 7
  - `add_new_option()` (*ats\_utilities.cli.json\_cli.JsonCLI* method), 8
  - `add_new_option()` (*ats\_utilities.cli.xml\_cli.XmlCLI* method), 9
  - `add_new_option()` (*ats\_utilities.cli.yaml\_cli.YamlCLI* method), 10
  - `add_operation()` (*ats\_utilities.option.ATSOptionParser* method), 46
  - `ATS_BASE_INFO` (*ats\_utilities.info.ATSInfo* attribute), 42
  - `ATS_BUILD_DATE` (*ats\_utilities.info.ATSInfo* attribute), 42
  - `ATS_CRITICAL` (*ats\_utilities.logging.ATSLogger* attribute), 45
  - `ATS_DEBUG` (*ats\_utilities.logging.ATSLogger* attribute), 45

ATS\_ERROR (*ats\_utilities.logging.ATSLogger attribute*), 45  
 ATS\_INFO (*ats\_utilities.logging.ATSLogger attribute*), 45  
 ats\_info\_ok (*ats\_utilities.info.ats\_info\_ok.ATSInfoOk attribute*), 38  
 ATS\_LICENCE (*ats\_utilities.info.ATSInfo attribute*), 42  
 ATS\_NAME (*ats\_utilities.info.ATSInfo attribute*), 42  
 ats\_utilities (*module*), 55  
 ats\_utilities.abstract (*module*), 3  
 ats\_utilities.checker (*module*), 4  
 ats\_utilities.cli (*module*), 10  
 ats\_utilities.cli.cfg\_cli (*module*), 6  
 ats\_utilities.cli.ini\_cli (*module*), 7  
 ats\_utilities.cli.json\_cli (*module*), 8  
 ats\_utilities.cli.xml\_cli (*module*), 8  
 ats\_utilities.cli.yaml\_cli (*module*), 9  
 ats\_utilities.config\_io (*module*), 26  
 ats\_utilities.config\_io.base\_check (*module*), 23  
 ats\_utilities.config\_io.base\_read (*module*), 25  
 ats\_utilities.config\_io.base\_write (*module*), 26  
 ats\_utilities.config\_io.cfg (*module*), 12  
 ats\_utilities.config\_io.cfg.cfg2object (*module*), 11  
 ats\_utilities.config\_io.cfg.object2cfg (*module*), 12  
 ats\_utilities.config\_io.ini (*module*), 15  
 ats\_utilities.config\_io.ini.ini2object (*module*), 13  
 ats\_utilities.config\_io.ini.object2ini (*module*), 14  
 ats\_utilities.config\_io.json (*module*), 17  
 ats\_utilities.config\_io.json.json2object (*module*), 16  
 ats\_utilities.config\_io.json.object2json (*module*), 17  
 ats\_utilities.config\_io.xml (*module*), 20  
 ats\_utilities.config\_io.xml.object2xml (*module*), 18  
 ats\_utilities.config\_io.xml.xml2object (*module*), 19  
 ats\_utilities.config\_io.yaml (*module*), 22  
 ats\_utilities.config\_io.yaml.object2yaml (*module*), 21  
 ats\_utilities.config\_io.yaml.yaml2object (*module*), 22  
 ats\_utilities.console\_io (*module*), 31  
 ats\_utilities.console\_io.error (*module*), 27  
 ats\_utilities.console\_io.success (*module*), 28  
 ats\_utilities.console\_io.verbose (*module*), 29  
 ats\_utilities.console\_io.warning (*module*), 30  
 ats\_utilities.cooperative (*module*), 31  
 ats\_utilities.exceptions (*module*), 36  
 ats\_utilities.exceptions.ats\_attribute\_error (*module*), 32  
 ats\_utilities.exceptions.ats\_bad\_call\_error (*module*), 32  
 ats\_utilities.exceptions.ats\_file\_error (*module*), 33  
 ats\_utilities.exceptions.ats\_key\_error (*module*), 33  
 ats\_utilities.exceptions.ats\_lookup\_error (*module*), 34  
 ats\_utilities.exceptions.ats\_parameter\_error (*module*), 34  
 ats\_utilities.exceptions.ats\_type\_error (*module*), 35  
 ats\_utilities.exceptions.ats\_value\_error (*module*), 35  
 ats\_utilities.final (*module*), 36  
 ats\_utilities.info (*module*), 41  
 ats\_utilities.info.ats\_build\_date (*module*), 37  
 ats\_utilities.info.ats\_info\_ok (*module*), 38  
 ats\_utilities.info.ats\_licence (*module*), 38  
 ats\_utilities.info.ats\_name (*module*), 39  
 ats\_utilities.info.ats\_version (*module*), 40  
 ats\_utilities.logging (*module*), 44  
 ats\_utilities.logging.ats\_logger\_file (*module*), 42  
 ats\_utilities.logging.ats\_logger\_name (*module*), 43  
 ats\_utilities.logging.ats\_logger\_status (*module*), 44  
 ats\_utilities.option (*module*), 45  
 ats\_utilities.register (*module*), 46  
 ats\_utilities.singleton (*module*), 49  
 ats\_utilities.singleton.base (*module*), 47  
 ats\_utilities.singleton.functional (*module*), 48  
 ats\_utilities.singleton.meta (*module*), 48  
 ats\_utilities.singleton.meta\_thread (*module*), 49  
 ats\_utilities.splash (*module*), 55  
 ats\_utilities.splash.ext\_infrastructure (*module*), 49  
 ats\_utilities.splash.github\_infrastructure (*module*), 50

ats\_utilities.splash.progress\_bar (module), 52

ats\_utilities.splash.splash\_property (module), 53

ats\_utilities.splash.terminal\_properties (module), 53

ATS\_VERSION (ats\_utilities.info.ATSInfo attribute), 42

ATS\_WARNING (ats\_utilities.logging.ATSLogger attribute), 45

ATSAttributeError, 32

ATSBadCallError, 33

ATSBuildDate (class in ats\_utilities.info.ats\_build\_date), 37

ATSChecker (class in ats\_utilities.checker), 4

ATSError, 36

ATSError (class in ats\_utilities.console\_io.error), 27

ATSFileError, 33

ATSFinal (class in ats\_utilities.final), 36

ATSInfo (class in ats\_utilities.info), 41

ATSInfoOk (class in ats\_utilities.info.ats\_info\_ok), 38

ATSKeyError, 34

ATSLicence (class in ats\_utilities.info.ats\_licence), 39

ATSLogger (class in ats\_utilities.logging), 44

ATSLoggerFile (class in ats\_utilities.logging.ats\_logger\_file), 42

ATSLoggerName (class in ats\_utilities.logging.ats\_logger\_name), 43

ATSLoggerStatus (class in ats\_utilities.logging.ats\_logger\_status), 44

ATSLookupError, 34

ATSName (class in ats\_utilities.info.ats\_name), 39

ATSOptionParser (class in ats\_utilities.option), 46

ATSParameterError, 35

ATSRegister (class in ats\_utilities.register), 47

ATSSuccess (class in ats\_utilities.console\_io.success), 28

ATSTypeError, 35

ATSValueError, 36

ATSVerbose (class in ats\_utilities.console\_io.verbose), 29

ATSVersion (class in ats\_utilities.info.ats\_version), 40

ATSWarning (class in ats\_utilities.console\_io.warning), 30

**B**

BaseReadConfig (class in ats\_utilities.config\_io.base\_read), 25

BaseWriteConfig (class in ats\_utilities.config\_io.base\_write), 26

build\_date (ats\_utilities.info.ats\_build\_date.ATSBuildDate attribute), 37

**C**

center () (ats\_utilities.splash.Splash method), 55

Cfg2Object (class in ats\_utilities.config\_io.cfg.cfg2object), 11

CfgBase (class in ats\_utilities.config\_io.cfg), 13

CfgCLI (class in ats\_utilities.cli.cfg\_cli), 6

check\_format () (ats\_utilities.config\_io.base\_check.FileChecking method), 24

check\_mode () (ats\_utilities.config\_io.base\_check.FileChecking method), 24

check\_params () (ats\_utilities.checker.ATSChecker method), 5

check\_path () (ats\_utilities.config\_io.base\_check.FileChecking method), 24

check\_types () (ats\_utilities.checker.ATSChecker method), 5

check\_values () (ats\_utilities.checker.ATSChecker method), 5

collect\_params () (ats\_utilities.checker.ATSChecker method), 5

ConfigFile (class in ats\_utilities.config\_io), 27

CooperativeMeta (class in ats\_utilities.cooperative), 32

**D**

DEFAULT\_BAR\_LENGTH (ats\_utilities.splash.progress\_bar.ProgressBar attribute), 52

DEFAULT\_CHAR\_OFF (ats\_utilities.splash.progress\_bar.ProgressBar attribute), 52

DEFAULT\_CHAR\_ON (ats\_utilities.splash.progress\_bar.ProgressBar attribute), 52

**E**

error\_message () (in module ats\_utilities.console\_io.error), 28

ExtInfrastructure (class in ats\_utilities.splash.ext\_infrastructure), 50

**F**

file\_path (ats\_utilities.config\_io.base\_read.BaseReadConfig attribute), 25

file\_path (ats\_utilities.config\_io.base\_write.BaseWriteConfig attribute), 26

FileChecking (class in ats\_utilities.config\_io.base\_check), 23

FORMAT\_ERROR (ats\_utilities.checker.ATSChecker attribute), 4

FunctionalSingleton () (in module ats\_utilities.singleton.functional), 48

get\_author\_text () (ats\_utilities.splash.ext\_infrastructure.ExtInfrastructure method), 50

get\_author\_text ()

(*ats\_utilities.splash.github\_infrastructure.GitHubInfrastructure* *JSON2Object* (class in *ats\_utilities.config\_io.json.json2object*), 16  
method), 51

get\_info\_text () (*ats\_utilities.splash.ext\_infrastructure.ExtInfrastructure* *JSONBase* (class in *ats\_utilities.config\_io.json*), 18  
method), 50

get\_info\_text () (*ats\_utilities.splash.github\_infrastructure.GitHubInfrastructure* *JsonCLI* (class in *ats\_utilities.cli.json\_cli*), 8  
method), 51

get\_issue\_text () (*ats\_utilities.splash.ext\_infrastructure.ExtInfrastructure* *License* (*ats\_utilities.info.ats\_licence.ATSLicence* at-  
tribute), 39  
method), 50

get\_issue\_text () (*ats\_utilities.splash.github\_infrastructure.GitHubInfrastructure* *LOG\_DATE\_FORMAT* (*ats\_utilities.logging.ATSLogger*  
attribute), 45  
method), 51

*GitHubInfrastructure* (class in *ats\_utilities.splash.github\_infrastructure*),  
51  
*LOG\_MSG\_FORMAT* (*ats\_utilities.logging.ATSLogger* at-  
tribute), 45  
*logger\_file* (*ats\_utilities.logging.ats\_logger\_file.ATSLoggerFile*  
attribute), 43

*logger\_name* (*ats\_utilities.logging.ats\_logger\_name.ATSLoggerName*  
attribute), 43

*logger\_status* (*ats\_utilities.logging.ats\_logger\_status.ATSLoggerStatu*  
attribute), 44

*Ini2Object* (class in *ats\_utilities.config\_io.ini.ini2object*), 14

*IniBase* (class in *ats\_utilities.config\_io.ini*), 15

*IniCLI* (class in *ats\_utilities.cli.ini\_cli*), 7

*is\_correct* () (*ats\_utilities.info.ATSInfo* method), 42

*is\_file\_ok* () (*ats\_utilities.config\_io.base\_check.FileChecking* *message* (*ats\_utilities.console\_io.error.ATSError* at-  
tribute), 28  
method), 24

*is\_not\_none* () (*ats\_utilities.config\_io.base\_read.BaseReadConfig* *message* (*ats\_utilities.console\_io.success.ATSSuccess*  
attribute), 29  
method), 25

*is\_not\_none* () (*ats\_utilities.config\_io.base\_write.BaseWriteConfig* *message* (*ats\_utilities.console\_io.verbose.ATSVerbose*  
attribute), 30  
method), 26

*is\_not\_none* () (*ats\_utilities.console\_io.error.ATSError* *message* (*ats\_utilities.console\_io.warning.ATSWarning*  
attribute), 31  
method), 28

*is\_not\_none* () (*ats\_utilities.console\_io.success.ATSSuccess* *MODES* (*ats\_utilities.config\_io.base\_check.FileChecking*  
attribute), 24  
method), 29

*is\_not\_none* () (*ats\_utilities.console\_io.verbose.ATSVerbose*  
method), 30

*is\_not\_none* () (*ats\_utilities.console\_io.warning.ATSWarning* *name* (*ats\_utilities.info.ats\_name.ATSName* attribute),  
40  
method), 31

*is\_not\_none* () (*ats\_utilities.info.ats\_build\_date.ATSBuildDate* *NO\_ERROR* (*ats\_utilities.checker.ATSChecker* attribute),  
4  
method), 38

*is\_not\_none* () (*ats\_utilities.info.ats\_licence.ATSLicence*  
method), 39

*is\_not\_none* () (*ats\_utilities.info.ats\_name.ATSName* *Object2Cfg* (class in *ats\_utilities.config\_io.cfg.object2cfg*), 12  
method), 40

*is\_not\_none* () (*ats\_utilities.info.ats\_version.ATSVersion* *Object2Ini* (class in *ats\_utilities.config\_io.ini.object2ini*), 14  
method), 41

*is\_tool\_ok* () (*ats\_utilities.config\_io.cfg.CfgBase* *Object2Json* (class in *ats\_utilities.config\_io.json.object2json*), 17  
method), 13

*is\_tool\_ok* () (*ats\_utilities.config\_io.ini.IniBase* *Object2Xml* (class in *ats\_utilities.config\_io.xml.object2xml*), 18  
method), 15

*is\_tool\_ok* () (*ats\_utilities.config\_io.json.JsonBase* *Object2Yaml* (class in *ats\_utilities.config\_io.yaml.object2yaml*),  
21  
method), 18

*is\_tool\_ok* () (*ats\_utilities.config\_io.xml.XmlBase*  
method), 20

*is\_tool\_ok* () (*ats\_utilities.config\_io.yaml.YamlBase*  
method), 23

## J

*JSON2Object* (class in *ats\_utilities.config\_io.json.json2object*), 16

*JSONBase* (class in *ats\_utilities.config\_io.json*), 18

*JsonCLI* (class in *ats\_utilities.cli.json\_cli*), 8

*License* (*ats\_utilities.info.ats\_licence.ATSLicence* at-  
tribute), 39

*LOG\_DATE\_FORMAT* (*ats\_utilities.logging.ATSLogger*  
attribute), 45

*LOG\_MSG\_FORMAT* (*ats\_utilities.logging.ATSLogger* at-  
tribute), 45

*logger\_file* (*ats\_utilities.logging.ats\_logger\_file.ATSLoggerFile*  
attribute), 43

*logger\_name* (*ats\_utilities.logging.ats\_logger\_name.ATSLoggerName*  
attribute), 43

*logger\_status* (*ats\_utilities.logging.ats\_logger\_status.ATSLoggerStatu*  
attribute), 44

## M

*message* (*ats\_utilities.console\_io.error.ATSError* at-  
tribute), 28

*message* (*ats\_utilities.console\_io.success.ATSSuccess*  
attribute), 29

*message* (*ats\_utilities.console\_io.verbose.ATSVerbose*  
attribute), 30

*message* (*ats\_utilities.console\_io.warning.ATSWarning*  
attribute), 31

*MODES* (*ats\_utilities.config\_io.base\_check.FileChecking*  
attribute), 24

## N

*name* (*ats\_utilities.info.ats\_name.ATSName* attribute),  
40

*NO\_ERROR* (*ats\_utilities.checker.ATSChecker* attribute),  
4

## O

*Object2Cfg* (class in *ats\_utilities.config\_io.cfg.object2cfg*), 12

*Object2Ini* (class in *ats\_utilities.config\_io.ini.object2ini*), 14

*Object2Json* (class in *ats\_utilities.config\_io.json.object2json*), 17

*Object2Xml* (class in *ats\_utilities.config\_io.xml.object2xml*), 18

*Object2Yaml* (class in *ats\_utilities.config\_io.yaml.object2yaml*),  
21

## P

*parse\_args* () (*ats\_utilities.cli.cfg\_cli.CfgCLI*  
method), 6



parse\_args () (ats\_utilities.cli.ini\_cli.IniCLI SingletonMeta (class in ats\_utilities.singleton.meta),  
 method), 7 48  
 parse\_args () (ats\_utilities.cli.json\_cli.JsonCLI size () (ats\_utilities.splash.terminal\_properties.TerminalProperties  
 method), 8 method), 54  
 parse\_args () (ats\_utilities.cli.xml\_cli.XmlCLI Splash (class in ats\_utilities.splash), 55  
 method), 9 SplashProperty (class in  
 parse\_args () (ats\_utilities.cli.yaml\_cli.YamlCLI ats\_utilities.splash.splash\_property), 53  
 method), 10 success\_message () (in module  
 parse\_args () (ats\_utilities.option.ATSOptionParser ats\_utilities.console\_io.success), 29  
 method), 46  
 plot\_progress () (ats\_utilities.splash.progress\_bar.ProgressBar T  
 method), 52 TerminalProperties (class in  
 priority\_error () (ats\_utilities.checker.ATSChecker ats\_utilities.splash.terminal\_properties),  
 method), 5 54  
 process (ats\_utilities.cli.cfg\_cli.CfgCLI attribute), 7 ThreadSafeSingletonMeta (class in  
 process (ats\_utilities.cli.ini\_cli.IniCLI attribute), 7 ats\_utilities.singleton.meta\_thread), 49  
 process (ats\_utilities.cli.json\_cli.JsonCLI attribute), 8 TYPE\_ERROR (ats\_utilities.checker.ATSChecker at-  
 process (ats\_utilities.cli.xml\_cli.XmlCLI attribute), 9 tribute), 4  
 process (ats\_utilities.cli.yaml\_cli.YamlCLI attribute),  
 10  
 ProgressBar (class in usage\_message () (ats\_utilities.checker.ATSChecker  
 ats\_utilities.splash.progress\_bar), 52 method), 5

**R**

read\_configuration (ats\_utilities.config\_io.base\_read.BaseReadConfig validate () (ats\_utilities.splash.splash\_property.SplashProperty  
 attribute), 25 method), 53  
 read\_configuration () (ats\_utilities.config\_io.cfg\_cfg2object.Cfg2Object VALUE\_ERROR (ats\_utilities.checker.ATSChecker  
 method), 11 attribute), 5  
 read\_configuration () (ats\_utilities.config\_io.ini\_ini2object.Ini2Object VERBOSE (ats\_utilities.cli.cfg\_cli.CfgCLI attribute), 6  
 method), 14 VERBOSE (ats\_utilities.cli.ini\_cli.IniCLI attribute), 7  
 read\_configuration () (ats\_utilities.config\_io.json\_json2object.Json2Object VERBOSE (ats\_utilities.cli.json\_cli.JsonCLI attribute), 8  
 method), 16 VERBOSE (ats\_utilities.cli.xml\_cli.XmlCLI attribute), 9  
 read\_configuration () (ats\_utilities.config\_io.xml\_xml2object.Xml2Object VERBOSE (ats\_utilities.cli.yaml\_cli.YamlCLI attribute),  
 method), 20 10  
 read\_configuration () (ats\_utilities.config\_io.json\_json2object.Json2Object VERBOSE (ats\_utilities.config\_io.base\_check.FileChecking  
 method), 16 attribute), 24  
 read\_configuration () (ats\_utilities.config\_io.xml\_xml2object.Xml2Object VERBOSE (ats\_utilities.config\_io.base\_read.BaseReadConfig  
 method), 20 attribute), 25  
 read\_configuration () (ats\_utilities.config\_io.yaml\_yaml2object.Yaml2Object VERBOSE (ats\_utilities.config\_io.base\_write.BaseWriteConfig  
 method), 22 attribute), 26  
 ROOT\_PACKAGE\_NAME (ats\_utilities.VerboseRoot at- VERBOSE (ats\_utilities.config\_io.cfg\_cfg2object.Cfg2Object  
 tribute), 56 attribute), 11

**S**

set\_and\_plot () (ats\_utilities.splash.progress\_bar.ProgressBar VERBOSE (ats\_utilities.config\_io.cfg.CfgBase attribute),  
 method), 52 13  
 set\_level () (ats\_utilities.splash.progress\_bar.ProgressBar VERBOSE (ats\_utilities.config\_io.cfg.object2cfg.Object2Cfg  
 method), 53 attribute), 12  
 show\_base\_info () (ats\_utilities.info.ATSInfo VERBOSE (ats\_utilities.config\_io.ConfigFile attribute),  
 method), 42 27  
 Singleton (class in ats\_utilities.singleton.base), 47 VERBOSE (ats\_utilities.config\_io.ini\_ini2object.Ini2Object  
 attribute), 14  
 VERBOSE (ats\_utilities.config\_io.ini.IniBase attribute),  
 15  
 VERBOSE (ats\_utilities.config\_io.ini.object2ini.Object2Ini  
 attribute), 15

VERBOSE (*ats\_utilities.config\_io.json.json2object.Json2Object* attribute), 16  
 VERBOSE (*ats\_utilities.config\_io.json.JsonBase* attribute), 18  
 VERBOSE (*ats\_utilities.config\_io.json.object2json.Object2Json* attribute), 17  
 VERBOSE (*ats\_utilities.config\_io.xml.object2xml.Object2Xml* attribute), 19  
 VERBOSE (*ats\_utilities.config\_io.xml.xml2object.Xml2Object* attribute), 20  
 VERBOSE (*ats\_utilities.config\_io.xml.XmlBase* attribute), 20  
 VERBOSE (*ats\_utilities.config\_io.yaml.object2yaml.Object2Yaml* attribute), 21  
 VERBOSE (*ats\_utilities.config\_io.yaml.yaml2object.Yaml2Object* attribute), 22  
 VERBOSE (*ats\_utilities.config\_io.yaml.YamlBase* attribute), 23  
 VERBOSE (*ats\_utilities.info.ats\_build\_date.ATSBuildDate* attribute), 37  
 VERBOSE (*ats\_utilities.info.ats\_info\_ok.ATSInfoOk* attribute), 38  
 VERBOSE (*ats\_utilities.info.ats\_licence.ATSLicence* attribute), 39  
 VERBOSE (*ats\_utilities.info.ats\_name.ATSName* attribute), 40  
 VERBOSE (*ats\_utilities.info.ats\_version.ATSVersion* attribute), 41  
 VERBOSE (*ats\_utilities.info.ATSInfo* attribute), 42  
 VERBOSE (*ats\_utilities.logging.ats\_logger\_file.ATSLoggerFile* attribute), 43  
 VERBOSE (*ats\_utilities.logging.ats\_logger\_name.ATSLoggerName* attribute), 43  
 VERBOSE (*ats\_utilities.logging.ats\_logger\_status.ATSLoggerStatus* attribute), 44  
 VERBOSE (*ats\_utilities.logging.ATSLogger* attribute), 45  
 VERBOSE (*ats\_utilities.option.ATSOptionParser* attribute), 46  
 VERBOSE (*ats\_utilities.splash.ext\_infrastructure.ExtInfrastructure* attribute), 50  
 VERBOSE (*ats\_utilities.splash.github\_infrastructure.GitHubInfrastructure* attribute), 51  
 VERBOSE (*ats\_utilities.splash.Splash* attribute), 55  
 VERBOSE (*ats\_utilities.splash.splash\_property.SplashProperty* attribute), 53  
 VERBOSE (*ats\_utilities.splash.terminal\_properties.TerminalProperties* attribute), 54  
 verbose\_message () (in module *ats\_utilities.console\_io.verbose*), 30  
 VerboseRoot (class in *ats\_utilities*), 56  
 version (*ats\_utilities.info.ats\_version.ATSVersion* attribute), 41

**W**  
 warning\_message () (in module *ats\_utilities.console\_io.warning*), 31  
 write\_configuration (*ats\_utilities.config\_io.base\_write.BaseWriteConfig* attribute), 26  
 write\_configuration () (*ats\_utilities.config\_io.cfg.object2cfg.Object2Cfg* method), 12  
 write\_configuration () (*ats\_utilities.config\_io.ini.object2ini.Object2Ini* method), 15  
 write\_configuration () (*ats\_utilities.config\_io.json.object2json.Object2Json* method), 17  
 write\_configuration () (*ats\_utilities.config\_io.xml.object2xml.Object2Xml* method), 19  
 write\_configuration () (*ats\_utilities.config\_io.yaml.object2yaml.Object2Yaml* method), 21  
 write\_log () (*ats\_utilities.logging.ATSLogger* method), 45

**X**  
 Xml2Object (class in *ats\_utilities.config\_io.xml.xml2object*), 19  
 XmlBase (class in *ats\_utilities.config\_io.xml*), 20  
 XmlCLI (class in *ats\_utilities.cli.xml\_cli*), 9

**Y**  
 Yaml2Object (class in *ats\_utilities.config\_io.yaml.yaml2object*), 22  
 YamlBase (class in *ats\_utilities.config\_io.yaml*), 23  
 YamlCLI (class in *ats\_utilities.cli.yaml\_cli*), 10