

Unittest for caching

December 21, 2020

Contents

1	Test Information	5
1.1	Test Candidate Information	5
1.2	Unittest Information	5
1.3	Test System Information	5
2	Statistic	5
2.1	Test-Statistic for testrun with python 2.7.18 (final)	5
2.2	Test-Statistic for testrun with python 3.8.5 (final)	6
2.3	Coverage Statistic	6
3	Testcases with no corresponding Requirement	7
3.1	Summary for testrun with python 2.7.18 (final)	7
3.1.1	caching.property_cache.json: Test cached data (full init)	7
3.1.2	caching.property_cache.json: Test cached data (partially init)	7
3.1.3	caching.property_cache.json: Test execution of save callback (full init)	7
3.1.4	caching.property_cache.json: Test full initialised JSON-Cache-Object	8
3.1.5	caching.property_cache.json: Test get from source caused by changed uid (full init)	8
3.1.6	caching.property_cache.json: Test get from source caused by changed uid (partially init)	8
3.1.7	caching.property_cache.json: Test get from source caused by increased data version (full init)	9
3.1.8	caching.property_cache.json: Test get from source caused by increased data version (partially init)	9
3.1.9	caching.property_cache.json: Test internal key usage	9
3.1.10	caching.property_cache.json: Test partially initialisation of JSON-Cache-Object	10
3.1.11	caching.property_cache.pickle: Test cached data (full init)	10
3.1.12	caching.property_cache.pickle: Test cached data (partially init)	11
3.1.13	caching.property_cache.pickle: Test execution of save callback (full init)	11
3.1.14	caching.property_cache.pickle: Test full initialised PICKLE-Cache-Object	11
3.1.15	caching.property_cache.pickle: Test get from source caused by changed uid (full init)	12
3.1.16	caching.property_cache.pickle: Test get from source caused by changed uid (partially init)	12
3.1.17	caching.property_cache.pickle: Test get from source caused by increased data version (full init)	12
3.1.18	caching.property_cache.pickle: Test get from source caused by increased data version (partially init)	13

3.1.19	caching.property_cache_pickle: Test internal key usage	13
3.1.20	caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object	14
3.2	Summary for testrun with python 3.8.5 (final)	14
3.2.1	caching.property_cache_json: Test cached data (full init)	14
3.2.2	caching.property_cache_json: Test cached data (partially init)	14
3.2.3	caching.property_cache_json: Test execution of save callback (full init)	15
3.2.4	caching.property_cache_json: Test full initialised JSON-Cache-Object	15
3.2.5	caching.property_cache_json: Test get from source caused by changed uid (full init)	15
3.2.6	caching.property_cache_json: Test get from source caused by changed uid (partially init)	16
3.2.7	caching.property_cache_json: Test get from source caused by increased data version (full init)	16
3.2.8	caching.property_cache_json: Test get from source caused by increased data version (partially init)	17
3.2.9	caching.property_cache_json: Test internal key usage	17
3.2.10	caching.property_cache_json: Test partially initialisation of JSON-Cache-Object	17
3.2.11	caching.property_cache_pickle: Test cached data (full init)	18
3.2.12	caching.property_cache_pickle: Test cached data (partially init)	18
3.2.13	caching.property_cache_pickle: Test execution of save callback (full init)	18
3.2.14	caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object	19
3.2.15	caching.property_cache_pickle: Test get from source caused by changed uid (full init)	19
3.2.16	caching.property_cache_pickle: Test get from source caused by changed uid (partially init)	20
3.2.17	caching.property_cache_pickle: Test get from source caused by increased data version (full init)	20
3.2.18	caching.property_cache_pickle: Test get from source caused by increased data version (partially init)	20
3.2.19	caching.property_cache_pickle: Test internal key usage	21
3.2.20	caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object	21

A Trace for testrun with python 2.7.18 (final) 22

A.1	Tests with status Warning (4)	22
A.1.1	caching.property_cache_json: Test full initialised JSON-Cache-Object	22
A.1.2	caching.property_cache_json: Test partially initialisation of JSON-Cache-Object	23
A.1.3	caching.property_cache_json: Test cached data (full init)	25
A.1.4	caching.property_cache_json: Test cached data (partially init)	26

A.2	Tests with status Info (16)	27
A.2.1	caching.property_cache_json: Test get from source caused by increased data version (full init)	27
A.2.2	caching.property_cache_json: Test get from source caused by increased data version (partially init)	28
A.2.3	caching.property_cache_json: Test get from source caused by changed uid (full init)	30
A.2.4	caching.property_cache_json: Test get from source caused by changed uid (partially init)	31
A.2.5	caching.property_cache_json: Test execution of save callback (full init)	32
A.2.6	caching.property_cache_json: Test internal key usage	33
A.2.7	caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object	34
A.2.8	caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object	34
A.2.9	caching.property_cache_pickle: Test cached data (full init)	35
A.2.10	caching.property_cache_pickle: Test cached data (partially init)	36
A.2.11	caching.property_cache_pickle: Test get from source caused by increased data version (full init)	37
A.2.12	caching.property_cache_pickle: Test get from source caused by increased data version (partially init)	38
A.2.13	caching.property_cache_pickle: Test get from source caused by changed uid (full init)	40
A.2.14	caching.property_cache_pickle: Test get from source caused by changed uid (partially init)	41
A.2.15	caching.property_cache_pickle: Test execution of save callback (full init)	42
A.2.16	caching.property_cache_pickle: Test internal key usage	43
B	Trace for testrun with python 3.8.5 (final)	44
B.1	Tests with status Info (20)	44
B.1.1	caching.property_cache_json: Test full initialised JSON-Cache-Object	44
B.1.2	caching.property_cache_json: Test partial initialisation of JSON-Cache-Object	44
B.1.3	caching.property_cache_json: Test cached data (full init)	45
B.1.4	caching.property_cache_json: Test cached data (partially init)	46
B.1.5	caching.property_cache_json: Test get from source caused by increased data version (full init)	47
B.1.6	caching.property_cache_json: Test get from source caused by increased data version (partially init)	48
B.1.7	caching.property_cache_json: Test get from source caused by changed uid (full init)	50
B.1.8	caching.property_cache_json: Test get from source caused by changed uid (partially init)	51
B.1.9	caching.property_cache_json: Test execution of save callback (full init)	52
B.1.10	caching.property_cache_json: Test internal key usage	53

Unittest for caching

B.1.11	caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object	54
B.1.12	caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object	54
B.1.13	caching.property_cache_pickle: Test cached data (full init)	55
B.1.14	caching.property_cache_pickle: Test cached data (partially init)	56
B.1.15	caching.property_cache_pickle: Test get from source caused by increased data version (full init) .	57
B.1.16	caching.property_cache_pickle: Test get from source caused by increased data version (partially init)	58
B.1.17	caching.property_cache_pickle: Test get from source caused by changed uid (full init)	60
B.1.18	caching.property_cache_pickle: Test get from source caused by changed uid (partially init)	61
B.1.19	caching.property_cache_pickle: Test execution of save callback (full init)	62
B.1.20	caching.property_cache_pickle: Test internal key usage	63
C	Test-Coverage	64
C.1	caching	64
C.1.1	caching.__init__.py	64

1 Test Information

1.1 Test Candidate Information

The Module `caching` is designed to store information in `json` or `pickle` files to support them much faster then generating them from the original source file. For more Information read the documentation.

Library Information

Name	caching
State	Released
Supported Interpreters	python2, python3
Version	6af573fff375a4ecd1843479e13e5c99

Dependencies

1.2 Unittest Information

Unittest Information

Version	e113c3759194c81e7df9e29b1debe00d
Testruns with	python 2.7.18 (final), python 3.8.5 (final)

1.3 Test System Information

System Information

Architecture	64bit
Distribution	Linux Mint 20 ulyana
Hostname	ahorn
Kernel	5.4.0-58-generic (#64-Ubuntu SMP Wed Dec 9 08:16:25 UTC 2020)
Machine	x86_64
Path	/user_data/data/dirk/prj/unittest/caching/unittest
System	Linux
Username	dirk

2 Statistic

2.1 Test-Statistic for testrun with python 2.7.18 (final)

Number of tests	20
Number of successfull tests	16
Number of possibly failed tests	4
Number of failed tests	0

Executionlevel	Full Test (all defined tests)
Time consumption	0.095s

2.2 Test-Statistic for testrun with python 3.8.5 (final)

Number of tests	20
Number of successfull tests	20
Number of possibly failed tests	0
Number of failed tests	0

Executionlevel	Full Test (all defined tests)
Time consumption	0.081s

2.3 Coverage Statistic

Module- or Filename	Line-Coverage	Branch-Coverage
caching	100.0%	100.0%
caching.__init__.py	100.0%	

3 Testcases with no corresponding Requirement

3.1 Summary for testrun with python 2.7.18 (final)

3.1.1 caching.property_cache_json: Test cached data (full init)

Testresult

This test was passed with the state: **Warning**. See also full trace in section A.1.3!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (31)
Start-Time:	2020-12-21 01:50:07,892
Finished-Time:	2020-12-21 01:50:07,895
Time-Consumption	0.003s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Info	Collecting data from cache instance.
Warning	Cached data is NOT correct. See detailed log for more information.

3.1.2 caching.property_cache_json: Test cached data (partially init)

Testresult

This test was passed with the state: **Warning**. See also full trace in section A.1.4!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (32)
Start-Time:	2020-12-21 01:50:07,896
Finished-Time:	2020-12-21 01:50:07,899
Time-Consumption	0.003s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Info	Collecting data from cache instance.
Warning	Cached data is NOT correct. See detailed log for more information.

3.1.3 caching.property_cache_json: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.5!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (37)
Start-Time:	2020-12-21 01:50:07,925
Finished-Time:	2020-12-21 01:50:07,925
Time-Consumption	0.001s

Testsummary:

Info	Installing save_callback, which sets a variable to True on execution.
Success	Save callback execution variable is correct (Content True and Type is <type 'bool'>).

3.1.4 caching.property_cache_json: Test full initialised JSON-Cache-Object

Testresult

This test was passed with the state: **Warning**. See also full trace in section A.1.1!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (29)
Start-Time:	2020-12-21 01:50:07,888
Finished-Time:	2020-12-21 01:50:07,890
Time-Consumption	0.002s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Info	Extracting storage object from property_cache_json for comparison.
Warning	Cache object is NOT correct. See detailed log for more information.

3.1.5 caching.property_cache_json: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.3!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (35)
Start-Time:	2020-12-21 01:50:07,912
Finished-Time:	2020-12-21 01:50:07,917
Time-Consumption	0.006s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Success	Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.6 caching.property_cache_json: Test get from source caused by changed uid (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.4!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (36)
Start-Time:	2020-12-21 01:50:07,918
Finished-Time:	2020-12-21 01:50:07,925

Time-Consumption 0.007s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=True).
Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.7 caching.property_cache.json: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.1!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/___init___py (33)
 Start-Time: 2020-12-21 01:50:07,899
 Finished-Time: 2020-12-21 01:50:07,902
 Time-Consumption 0.003s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=True).
Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.8 caching.property_cache.json: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.2!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/___init___py (34)
 Start-Time: 2020-12-21 01:50:07,902
 Finished-Time: 2020-12-21 01:50:07,911
 Time-Consumption 0.009s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=True).
Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.9 caching.property_cache.json: Test internal key usage

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.6!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (38)
 Start-Time: 2020-12-21 01:50:07,926
 Finished-Time: 2020-12-21 01:50:07,933
 Time-Consumption 0.007s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=True).
Info Extracting storage object from property_cache.json for comparison.
Success Cache is correct (Content {u'__property_cache_data_version_': u'no data version', u'__property_cache_data_version_': u'no second data version', u'__property_cache_uid_': u'no uid', u'__property_cache_uid_': u'no second uid'}) and Type is <type 'dict'>).
Success Keyfilter returnvalue for 5 (<type 'int'>) is correct (Content 5 and Type is <type 'int'>).

3.1.10 caching.property_cache_json: Test partially initialisation of JSON-Cache-Object

Testresult

This test was passed with the state: **Warning**. See also full trace in section A.1.2!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (30)
 Start-Time: 2020-12-21 01:50:07,890
 Finished-Time: 2020-12-21 01:50:07,892
 Time-Consumption 0.002s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=False).
Info Partially initialising cache object by requesting some information.
Info Extracting storage object from property_cache.json for comparison.
Warning Cache object is NOT correct. See detailed log for more information.

3.1.11 caching.property_cache_pickle: Test cached data (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.9!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (44)
 Start-Time: 2020-12-21 01:50:07,952
 Finished-Time: 2020-12-21 01:50:07,958
 Time-Consumption 0.006s

Testsummary:

Info Initialising property_cache.pickle (load_all_on_init=True).
Info Collecting data from cache instance.

Success Cached data is correct (Content {'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3', '2': 'two', '4': 4}, 'str': 'string', 'integer': 17} and Type is <type 'dict'>).

3.1.12 caching.property_cache_pickle: Test cached data (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.10!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (45)
Start-Time:	2020-12-21 01:50:07,958
Finished-Time:	2020-12-21 01:50:07,967
Time-Consumption	0.009s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
Info	Collecting data from cache instance.
Success	Cached data is correct (Content {'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3', '2': 'two', '4': 4}, 'str': 'string', 'integer': 17} and Type is <type 'dict'>).

3.1.13 caching.property_cache_pickle: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.15!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (50)
Start-Time:	2020-12-21 01:50:07,988
Finished-Time:	2020-12-21 01:50:07,988
Time-Consumption	0.000s

Testsummary:

Info	Installing save_callback, which sets a variable to True on execution.
Success	Save callback execution variable is correct (Content True and Type is <type 'bool'>).

3.1.14 caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.7!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (42)
Start-Time:	2020-12-21 01:50:07,933
Finished-Time:	2020-12-21 01:50:07,939

Time-Consumption 0.006s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Info Extracting storage object from property_cache_pickle for comparison.
Success Cache object is correct (Content {'str': 'string', 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3', '2': 'two', '4': 4}, 'unicode': u'unicode', 'integer': 17} and Type is <type 'dict'>).

3.1.15 caching.property_cache_pickle: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.13!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (48)
 Start-Time: 2020-12-21 01:50:07,979
 Finished-Time: 2020-12-21 01:50:07,982
 Time-Consumption 0.003s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.16 caching.property_cache_pickle: Test get from source caused by changed uid (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.14!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (49)
 Start-Time: 2020-12-21 01:50:07,983
 Finished-Time: 2020-12-21 01:50:07,987
 Time-Consumption 0.005s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.17 caching.property_cache_pickle: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.11!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (46)
 Start-Time: 2020-12-21 01:50:07,968
 Finished-Time: 2020-12-21 01:50:07,971
 Time-Consumption 0.003s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.18 caching.property_cache_pickle: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.12!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (47)
 Start-Time: 2020-12-21 01:50:07,971
 Finished-Time: 2020-12-21 01:50:07,978
 Time-Consumption 0.007s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

3.1.19 caching.property_cache_pickle: Test internal key usage

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.16!

Testrun: python 2.7.18 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (51)
 Start-Time: 2020-12-21 01:50:07,988
 Finished-Time: 2020-12-21 01:50:07,990
 Time-Consumption 0.002s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Info Extracting storage object from property_cache_pickle for comparison.
Success Cache is correct (Content {'__property_cache_data_version_': u'no data version', '__property_cache_data_version_': u'no second data version', '__property_cache_uid_': u'no uid', '__property_cache_uid_': u'no second uid'}) and Type is <type 'dict'>).
Success Keyfilter returnvalue for 5 (<type 'int'>) is correct (Content 5 and Type is <type 'int'>).

3.1.20 caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**. See also full trace in section A.2.8!

Testrun:	python 2.7.18 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (43)
Start-Time:	2020-12-21 01:50:07,940
Finished-Time:	2020-12-21 01:50:07,951
Time-Consumption	0.011s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=False).
Info	Partially initialising cache object by requesting some information.
Info	Extracting storage object from property_cache_pickle for comparison.
Success	Cache object is correct (Content {'float': 3.14159, 'list': [1, 'two', '3', 4], 'str': 'string'} and Type is <type 'dict'>).

3.2 Summary for testrun with python 3.8.5 (final)

3.2.1 caching.property_cache_json: Test cached data (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.3!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (31)
Start-Time:	2020-12-21 01:50:08,687
Finished-Time:	2020-12-21 01:50:08,689
Time-Consumption	0.002s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Info	Collecting data from cache instance.
Success	Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'}) and Type is <class 'dict'>).

3.2.2 caching.property_cache_json: Test cached data (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.4!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (32)
Start-Time:	2020-12-21 01:50:08,689
Finished-Time:	2020-12-21 01:50:08,693

Time-Consumption 0.004s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=True).
Info Collecting data from cache instance.
Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'}) and Type is <class 'dict'>).

3.2.3 caching.property_cache.json: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.9!

Testrun: python 3.8.5 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (37)
 Start-Time: 2020-12-21 01:50:08,734
 Finished-Time: 2020-12-21 01:50:08,734
 Time-Consumption 0.000s

Testsummary:

Info Installing save_callback, which sets a variable to True on execution.
Success Save callback execution variable is correct (Content True and Type is <class 'bool'>).

3.2.4 caching.property_cache.json: Test full initialised JSON-Cache-Object

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.1!

Testrun: python 3.8.5 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (29)
 Start-Time: 2020-12-21 01:50:08,674
 Finished-Time: 2020-12-21 01:50:08,679
 Time-Consumption 0.005s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=True).
Info Extracting storage object from property_cache.json for comparison.
Success Cache object is correct (Content {'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'float': 3.14159, 'integer': 17, 'list': [1, 'two', '3', 4], 'str': 'string', 'unicode': 'unicode'}) and Type is <class 'dict'>).

3.2.5 caching.property_cache.json: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.7!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (35)
Start-Time:	2020-12-21 01:50:08,708
Finished-Time:	2020-12-21 01:50:08,722
Time-Consumption	0.014s

Testsummary:

Info	Initialising property_cache.json (load_all_on_init=True).
Success	Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.6 caching.property_cache.json: Test get from source caused by changed uid (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.8!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (36)
Start-Time:	2020-12-21 01:50:08,724
Finished-Time:	2020-12-21 01:50:08,733
Time-Consumption	0.010s

Testsummary:

Info	Initialising property_cache.json (load_all_on_init=True).
Success	Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.7 caching.property_cache.json: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.5!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (33)
Start-Time:	2020-12-21 01:50:08,693
Finished-Time:	2020-12-21 01:50:08,700
Time-Consumption	0.007s

Testsummary:

Info	Initialising property_cache.json (load_all_on_init=True).
Success	Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.8 caching.property_cache_json: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.6!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (34)
Start-Time:	2020-12-21 01:50:08,700
Finished-Time:	2020-12-21 01:50:08,707
Time-Consumption	0.007s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Success	Instance data after increasing data_version is correct (Content {'str': 'string_', 'unicode': 'unicode_', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.9 caching.property_cache_json: Test internal key usage

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.10!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (38)
Start-Time:	2020-12-21 01:50:08,734
Finished-Time:	2020-12-21 01:50:08,735
Time-Consumption	0.001s

Testsummary:

Info	Initialising property_cache_json (load_all_on_init=True).
Info	Extracting storage object from property_cache_json for comparison.
Success	Cache is correct (Content {'__property_cache_data_version_': 'no second data version', '__property_cache_uid_': 'no second uid', '__property_cache_data_version_': 'no data version', '__property_cache_uid_': 'no uid'} and Type is <class 'dict'>).
Success	Keyfilter returnvalue for 5 (<class 'int'>) is correct (Content 5 and Type is <class 'int'>).

3.2.10 caching.property_cache_json: Test partially initialisation of JSON-Cache-Object

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.2!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (30)
Start-Time:	2020-12-21 01:50:08,679
Finished-Time:	2020-12-21 01:50:08,686
Time-Consumption	0.007s

Testsummary:

Info Initialising property_cache.json (load_all_on_init=False).
Info Partially initialising cache object by requesting some information.
Info Extracting storage object from property_cache.json for comparison.
Success Cache object is correct (Content {'integer': 17, 'str': 'string', 'unicode': 'unicode'} and Type is <class 'dict'>).

3.2.11 caching.property_cache_pickle: Test cached data (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.13!

Testrun: python 3.8.5 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (44)
 Start-Time: 2020-12-21 01:50:08,738
 Finished-Time: 2020-12-21 01:50:08,740
 Time-Consumption 0.002s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Info Collecting data from cache instance.
Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'} and Type is <class 'dict'>).

3.2.12 caching.property_cache_pickle: Test cached data (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.14!

Testrun: python 3.8.5 (final)
 Caller: /user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (45)
 Start-Time: 2020-12-21 01:50:08,740
 Finished-Time: 2020-12-21 01:50:08,741
 Time-Consumption 0.001s

Testsummary:

Info Initialising property_cache_pickle (load_all_on_init=True).
Info Collecting data from cache instance.
Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'} and Type is <class 'dict'>).

3.2.13 caching.property_cache_pickle: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.19!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (50)
Start-Time:	2020-12-21 01:50:08,757
Finished-Time:	2020-12-21 01:50:08,758
Time-Consumption	0.001s

Testsummary:

Info	Installing save_callback, which sets a variable to True on execution.
Success	Save callback execution variable is correct (Content True and Type is <class 'bool'>).

3.2.14 caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.11!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (42)
Start-Time:	2020-12-21 01:50:08,736
Finished-Time:	2020-12-21 01:50:08,737
Time-Consumption	0.001s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
Info	Extracting storage object from property_cache_pickle for comparison.
Success	Cache object is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}} and Type is <class 'dict'>).

3.2.15 caching.property_cache_pickle: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.17!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (48)
Start-Time:	2020-12-21 01:50:08,748
Finished-Time:	2020-12-21 01:50:08,751
Time-Consumption	0.003s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
Success	Instance data after changing uid is correct (Content {'_string_': '_string_', 'unicode': '_unicode_', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.16 caching.property_cache_pickle: Test get from source caused by changed uid (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.18!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (49)
Start-Time:	2020-12-21 01:50:08,752
Finished-Time:	2020-12-21 01:50:08,757
Time-Consumption	0.005s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
Success	Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.17 caching.property_cache_pickle: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.15!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (46)
Start-Time:	2020-12-21 01:50:08,741
Finished-Time:	2020-12-21 01:50:08,742
Time-Consumption	0.001s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
Success	Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.18 caching.property_cache_pickle: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.16!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init....py (47)
Start-Time:	2020-12-21 01:50:08,742
Finished-Time:	2020-12-21 01:50:08,748
Time-Consumption	0.006s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
-------------	---

Success Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

3.2.19 caching.property_cache_pickle: Test internal key usage

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.20!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (51)
Start-Time:	2020-12-21 01:50:08,758
Finished-Time:	2020-12-21 01:50:08,761
Time-Consumption	0.002s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=True).
Info	Extracting storage object from property_cache_pickle for comparison.
Success	Cache is correct (Content {'__property_cache_uid_': 'no uid', '___property_cache_uid_': 'no second uid', '__property_cache_data_version_': 'no data version', '___property_cache_data_version_': 'no second data version'} and Type is <class 'dict'>).
Success	Keyfilter returnvalue for 5 (<class 'int'>) is correct (Content 5 and Type is <class 'int'>).

3.2.20 caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**. See also full trace in section B.1.12!

Testrun:	python 3.8.5 (final)
Caller:	/user_data/data/dirk/prj/unittest/caching/unittest/src/tests/_init_.py (43)
Start-Time:	2020-12-21 01:50:08,737
Finished-Time:	2020-12-21 01:50:08,738
Time-Consumption	0.001s

Testsummary:

Info	Initialising property_cache_pickle (load_all_on_init=False).
Info	Partially initialising cache object by requesting some information.
Info	Extracting storage object from property_cache_pickle for comparison.
Success	Cache object is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17} and Type is <class 'dict'>).

A Trace for testrun with python 2.7.18 (final)

A.1 Tests with status **Warning** (4)

A.1.1 caching.property_cache_json: Test full initialised JSON-Cache-Object

Testresult

This test was passed with the state: **Warning**.

Info Initialising property_cache_json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.json as cache
 ↪ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↪ 'list']

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.json)

Info Extracting storage object from property_cache_json for comparison.

JsonCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.json)

Using storage object of cache class for comparison: {u'str': u'string',

↪ u'_property_cache_uid': u'my_unique_id', u'float': 3.14159, u'list': [1, u'two', u'3',
 ↪ 4], u'dict': {u'1': 1, u'3': u'3', u'2': u'two', u'4': 4}, u'unicode': u'unicode',
 ↪ u'_property_cache_data_version_': 1, u'integer': 17}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Warning Cache object is NOT correct. See detailed log for more information.

```
Result (Cache object): { u'str': u'string', u'float': 3.14159, u'list': [ 1, u'two', u'3', 4
↳ ], u'dict': { u'1': 1, u'3': u'3', u'2': u'two', u'4': 4 }, u'unicode': u'unicode',
↳ u'integer': 17 } (<type 'dict'>)
```

```
Expectation (Cache object): result = { 'dict': { '1': 1, '3': '3', '2': 'two', '4': 4 },
↳ 'unicode': u'unicode', 'str': 'string', 'integer': 17, 'float': 3.14159, 'list': [ 1,
↳ 'two', '3', 4 ] } (<type 'dict'>)
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.str (u'string'). Check for known json
↳ storage type deviation <str> -> <unicode>.
```

```
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
```

```
Content u'two' is incorrect for test_variable.
```

```
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
```

```
Content u'two' is incorrect for test_variable.
```

```
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'two').
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.list[1] (u'two'). Check for known json
↳ storage type deviation <str> -> <unicode>.
```

```
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'3').
```

```
Content u'3' is incorrect for test_variable.
```

```
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'3').
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.list[2] (u'3'). Check for known json
↳ storage type deviation <str> -> <unicode>.
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.dict.3 (u'3'). Check for known json
↳ storage type deviation <str> -> <unicode>.
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.dict.2 (u'two'). Check for known json
↳ storage type deviation <str> -> <unicode>.
```

A.1.2 caching.property_cache_json: Test partially initialisation of JSON-Cache-Object

Testresult

This test was passed with the state: **Warning**.

Info Initialising property_cache_json (load_all_on_init=False).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

```
↳ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json as
↳ cache file.
```

```
JsonCache: Cache file does not exists (yet).
```

```
JsonCache: cache-file stored
```

```
↳ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)
```

Info Partially initialising cache object by requesting some information.

Unittest for caching

```
JsonCache: Loading property for 'float' from source instance (3.14159)
```

```
JsonCache: cache-file stored
```

```
↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)
```

```
JsonCache: Loading property for 'list' from source instance ([1, 'two', '3', 4])
```

```
JsonCache: cache-file stored
```

```
↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)
```

```
JsonCache: Loading property for 'str' from source instance ('string')
```

```
JsonCache: cache-file stored
```

```
↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)
```

Info Extracting storage object from property_cache.json for comparison.

```
JsonCache: Loading properties from cache
```

```
↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)
```

```
Using storage object of cache class for comparison: {u'_property_cache_data_version': 1,  
↪ u'float': 3.14159, u'list': [1, u'two', u'3', 4], u'str': u'string',  
↪ u'_property_cache_uid_': u'my_unique_id'}
```

```
Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_
```

Warning Cache object is NOT correct. See detailed log for more information.

```
Result (Cache object): { u'float': 3.14159, u'list': [ 1, u'two', u'3', 4 ], u'str':  
↪ u'string' } (<type 'dict'>)
```

```
Expectation (Cache object): result = { 'str': 'string', 'float': 3.14159, 'list': [ 1, 'two',  
↪ '3', 4 ] } (<type 'dict'>)
```

```
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
```

```
Content u'two' is incorrect for test_variable.
```

```
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
```

```
Content u'two' is incorrect for test_variable.
```

```
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'two').
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.list[1] (u'two'). Check for known json  
↪ storage type deviation <str> -> <unicode>.
```

```
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'3').
```

```
Content u'3' is incorrect for test_variable.
```

```
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'3').
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.list[2] (u'3'). Check for known json  
↪ storage type deviation <str> -> <unicode>.
```

```
Type <type 'unicode'> is NOT <type 'str'> for result.str (u'string'). Check for known json  
↪ storage type deviation <str> -> <unicode>.
```

A.1.3 caching.property_cache_json: Test cached data (full init)

Testresult

This test was passed with the state: **Warning**.

Info Initialising property_cache_json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /cache_data_test_load_on_init.json as cache
 ↵ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↵ 'list']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ cache_data_test_load_on_init.json)

Info Collecting data from cache instance.

JsonCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/cache_data_test_load_on_init.json)

JsonCache: Providing property for 'dict' from cache ({u'1': 1, u'3': u'3', u'2': u'two',
 ↵ u'4': 4})

JsonCache: Providing property for 'unicode' from cache (u'unicode')

JsonCache: Providing property for 'str' from cache (u'string')

JsonCache: Providing property for 'integer' from cache (17)

JsonCache: Providing property for 'float' from cache (3.14159)

JsonCache: Providing property for 'list' from cache ([1, u'two', u'3', 4])

Warning Cached data is NOT correct. See detailed log for more information.

```

Result (Cached data): { 'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float':
↳ 3.14159, 'list': [ 1, u'two', u'3', 4 ], 'dict': { u'1': 1, u'3': u'3', u'2': u'two',
↳ u'4': 4 }, 'str': u'string', 'integer': 17 } (<type 'dict'>)
Expectation (Cached data): result = { 'uncached': 'uncached_data_of_class', 'dict': { '1': 1,
↳ '3': '3', '2': 'two', '4': 4 }, 'unicode': u'unicode', 'str': 'string', 'integer': 17,
↳ 'float': 3.14159, 'list': [ 1, 'two', '3', 4 ] } (<type 'dict'>)
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
Content u'two' is incorrect for test_variable.
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
Content u'two' is incorrect for test_variable.
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'two').
Type <type 'unicode'> is NOT <type 'str'> for result.list[1] (u'two'). Check for known json
↳ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'3').
Content u'3' is incorrect for test_variable.
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'3').
Type <type 'unicode'> is NOT <type 'str'> for result.list[2] (u'3'). Check for known json
↳ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for result.dict.3 (u'3'). Check for known json
↳ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for result.dict.2 (u'two'). Check for known json
↳ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for result.str (u'string'). Check for known json
↳ storage type deviation <str> -> <unicode>.

```

A.1.4 caching.property_cache_json: Test cached data (partially init)

Testresult

This test was passed with the state: **Warning**.

Info Initialising property_cache_json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

```

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data_
↳ /cache_data_test_no_load_on_init.json as cache
↳ file.

```

```

JsonCache: Cache file does not exists (yet).

```

```

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
↳ 'list']

```

```

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ cache_data_test_no_load_on_init.json)

```

Info Collecting data from cache instance.

```

JsonCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.
JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↪ output_data/cache_data_test_no_load_on_init.json)
JsonCache: Providing property for 'dict' from cache ({u'1': 1, u'3': u'3', u'2': u'two',
↪ u'4': 4})
JsonCache: Providing property for 'unicode' from cache (u'unicode')
JsonCache: Providing property for 'str' from cache (u'string')
JsonCache: Providing property for 'integer' from cache (17)
JsonCache: Providing property for 'float' from cache (3.14159)
JsonCache: Providing property for 'list' from cache ([1, u'two', u'3', 4])

```

Warning Cached data is NOT correct. See detailed log for more information.

```

Result (Cached data): { 'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float':
↪ 3.14159, 'list': [ 1, u'two', u'3', 4 ], 'dict': { u'1': 1, u'3': u'3', u'2': u'two',
↪ u'4': 4 }, 'str': u'string', 'integer': 17 } (<type 'dict'>)
Expectation (Cached data): result = { 'uncached': 'uncached_data_of_class', 'dict': { '1': 1,
↪ '3': '3', '2': 'two', '4': 4 }, 'unicode': u'unicode', 'str': 'string', 'integer': 17,
↪ 'float': 3.14159, 'list': [ 1, 'two', '3', 4 ] } (<type 'dict'>)
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
Content u'two' is incorrect for test_variable.
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'two').
Content u'two' is incorrect for test_variable.
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'two').
Type <type 'unicode'> is NOT <type 'str'> for result.list[1] (u'two'). Check for known json
↪ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for test_variable (u'3').
Content u'3' is incorrect for test_variable.
Type <type 'unicode'> is NOT <type 'int'> for test_variable (u'3').
Type <type 'unicode'> is NOT <type 'str'> for result.list[2] (u'3'). Check for known json
↪ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for result.dict.3 (u'3'). Check for known json
↪ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for result.dict.2 (u'two'). Check for known json
↪ storage type deviation <str> -> <unicode>.
Type <type 'unicode'> is NOT <type 'str'> for result.str (u'string'). Check for known json
↪ storage type deviation <str> -> <unicode>.

```

A.2 Tests with status Info (16)

A.2.1 caching.property_cache.json: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /data_version_test_load_on_init.json as cache
 ↵ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↵ 'list']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ data_version_test_load_on_init.json)

Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828,
 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and
 Type is <type 'dict'>).

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/data_version_test_load_on_init.json)

JsonCache: Data version increased, ignoring previous cache data

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↵ 'list']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ data_version_test_load_on_init.json)

JsonCache: Providing property for 'dict' from cache ({'1': '1', '3': 'three', '2': 2, '4':
 ↵ '4'})

JsonCache: Providing property for 'unicode' from cache (u'__unicode__')

JsonCache: Providing property for 'str' from cache ('__string__')

JsonCache: Providing property for 'integer' from cache (34)

JsonCache: Providing property for 'float' from cache (2.71828)

JsonCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

Result (Instance data after increasing data_version): { 'unicode': u'__unicode__', 'float':
 ↵ 2.71828, 'list': ['one', 2, 3, '4'], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4'
 ↵ }, 'str': '__string__', 'integer': 34 } (<type 'dict'>)

Expectation (Instance data after increasing data_version): result = { 'dict': { '1': '1',
 ↵ '3': 'three', '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__',
 ↵ 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'] } (<type 'dict'>)

A.2.2 caching.property_cache.json: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Unittest for caching

Deleting cache file from filesystem to ensure identical conditions for each test run.

```
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ /data_version_test_no_load_on_init.json as cache  
↳ file.
```

```
JsonCache: Cache file does not exists (yet).
```

```
JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',  
↳ 'list']
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

```
JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/  
↳ output_data/data_version_test_no_load_on_init.json)
```

```
JsonCache: Data version increased, ignoring previous cache data
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'dict' from source instance ({'1': '1', '3': 'three', '2': 2,  
↳ '4': '4'})
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'unicode' from source instance (u'__unicode__')
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'str' from source instance ('__string__')
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'integer' from source instance (34)
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'float' from source instance (2.71828)
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↳ data_version_test_no_load_on_init.json)
```

```
Result (Instance data after increasing data_version): { 'unicode': u'__unicode__', 'float':  
↳ 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4'  
↳ }, 'str': '__string__', 'integer': 34 } (<type 'dict'>)
```

```
Expectation (Instance data after increasing data_version): result = { 'dict': { '1': '1',  
↳ '3': 'three', '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__',  
↳ 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ] } (<type 'dict'>)
```

A.2.3 caching.property_cache.json: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.json
 ↪ as cache file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↪ 'list']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
 ↪ uid_test_load_on_init.json)

Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
 ↪ output_data/uid_test_load_on_init.json)

JsonCache: Source uid changed, ignoring previous cache data

JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↪ 'list']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
 ↪ uid_test_load_on_init.json)

JsonCache: Providing property for 'dict' from cache ({'1': '1', '3': 'three', '2': 2, '4':
 ↪ '4'})

JsonCache: Providing property for 'unicode' from cache (u'__unicode__')

JsonCache: Providing property for 'str' from cache ('__string__')

JsonCache: Providing property for 'integer' from cache (34)

JsonCache: Providing property for 'float' from cache (2.71828)

JsonCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

Result (Instance data after changing uid): { 'unicode': u'__unicode__', 'float': 2.71828,
 ↪ 'list': ['one', 2, 3, '4'], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4' },
 ↪ 'str': '__string__', 'integer': 34 } (<type 'dict'>)

Expectation (Instance data after changing uid): result = { 'dict': { '1': '1', '3': 'three',
 ↪ '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__', 'integer': 34,
 ↪ 'float': 2.71828, 'list': ['one', 2, 3, '4'] } (<type 'dict'>)

A.2.4 caching.property_cache.json: Test get from source caused by changed uid (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

```
Deleting cache file from filesystem to ensure identical conditions for each test run.
```

```
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data_
↪ /uid_test_no_load_on_init.json as cache
↪ file.
```

```
JsonCache: Cache file does not exists (yet).
```

```
JsonCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
↪ 'list']
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
```

Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '.__string__', 'integer': 34} and Type is <type 'dict'>).

```

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↳ output_data/uid_test_no_load_on_init.json)
JsonCache: Source uid changed, ignoring previous cache data
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'dict' from source instance ({'1': '1', '3': 'three', '2': 2,
↳ '4': '4'})
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'unicode' from source instance (u'__unicode__')
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'str' from source instance ('__string__')
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'integer' from source instance (34)
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'float' from source instance (2.71828)
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.json)
Result (Instance data after changing uid): { 'unicode': u'__unicode__', 'float': 2.71828,
↳ 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4' },
↳ 'str': '__string__', 'integer': 34 } (<type 'dict'>)
Expectation (Instance data after changing uid): result = { 'dict': { '1': '1', '3': 'three',
↳ '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__', 'integer': 34,
↳ 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ] } (<type 'dict'>)

```

A.2.5 caching.property_cache.json: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**.

Info Installing save_callback, which sets a variable to True on execution.

Success Save callback execution variable is correct (Content True and Type is <type 'bool'>).

```

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ save_callback_load_on_init.json)
Result (Save callback execution variable): True (<type 'bool'>)
Expectation (Save callback execution variable): result = True (<type 'bool'>)

```

A.2.6 caching.property_cache_json: Test internal key usage

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.json as
 ↪ cache file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['__property_cache_data_version',
 ↪ '_property_cache_data_version', '__property_cache_uid', '_property_cache_uid']

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.json)

Info Extracting storage object from property_cache_json for comparison.

JsonCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.json)

Using storage object of cache class for comparison: {u'__property_cache_data_version': u'no

↪ data version', u'___property_cache_data_version': u'no second data version',
 ↪ u'__property_cache_uid': u'no uid', u'_property_cache_uid': u'my_unique_id',
 ↪ u'_property_cache_data_version': 1, u'___property_cache_uid': u'no second uid'}

Deleting overhead keys: _property_cache_data_version, _property_cache_uid

Success Cache is correct (Content {u'__property_cache_data_version': u'no data version',
 u'___property_cache_data_version': u'no second data version', u'__property_cache_uid': u'no uid',
 u'___property_cache_uid': u'no second uid'} and Type is <type 'dict'>).

Result (Cache): { u'__property_cache_data_version': u'no data version',

↪ u'___property_cache_data_version': u'no second data version', u'__property_cache_uid':
 ↪ u'no uid', u'___property_cache_uid': u'no second uid' } (<type 'dict'>)

Expectation (Cache): result = { '__property_cache_data_version': u'no data version',

↪ '___property_cache_data_version': u'no second data version', '__property_cache_uid':
 ↪ u'no uid', '___property_cache_uid': u'no second uid' } (<type 'dict'>)

Success Keyfilter returnvalue for 5 (<type 'int'>) is correct (Content 5 and Type is <type 'int'>).

Result (Keyfilter returnvalue for 5 (<type 'int'>)): 5 (<type 'int'>)

Expectation (Keyfilter returnvalue for 5 (<type 'int'>)): result = 5 (<type 'int'>)

A.2.7 caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.pkl as cache
 ↪ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↪ 'list']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.pkl)

Info Extracting storage object from property_cache_pickle for comparison.

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.pkl)

Using storage object of cache class for comparison: {'str': 'string', '_property_cache_uid':

↪ 'my_unique_id', 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3',
 ↪ '2': 'two', '4': 4}, 'unicode': u'unicode', '_property_cache_data_version': 1,
 ↪ 'integer': 17}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache object is correct (Content {'str': 'string', 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3', '2': 'two', '4': 4}, 'unicode': u'unicode', 'integer': 17} and Type is <type 'dict'>).

Result (Cache object): { 'str': 'string', 'float': 3.14159, 'list': [1, 'two', '3', 4],

↪ 'dict': { '1': 1, '3': '3', '2': 'two', '4': 4 }, 'unicode': u'unicode', 'integer': 17 }
 ↪ (<type 'dict'>)

Expectation (Cache object): result = { 'dict': { '1': 1, '3': '3', '2': 'two', '4': 4 },

↪ 'unicode': u'unicode', 'str': 'string', 'integer': 17, 'float': 3.14159, 'list': [1,
 ↪ 'two', '3', 4] } (<type 'dict'>)

A.2.8 caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=False).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl as
 ↪ cache file.

PickCache: Cache file does not exists (yet).

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

Info Partially initialising cache object by requesting some information.

PickCache: Loading property for 'float' from source instance (3.14159)

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

PickCache: Loading property for 'list' from source instance ([1, 'two', '3', 4])

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

PickCache: Loading property for 'str' from source instance ('string')

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

Info Extracting storage object from property_cache_pickle for comparison.

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

Using storage object of cache class for comparison: {'_property_cache_data_version': 1,

↪ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'str': 'string', '_property_cache_uid':

↪ 'my_unique_id'}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache object is correct (Content {'float': 3.14159, 'list': [1, 'two', '3', 4], 'str': 'string'} and Type is <type 'dict'>).

Result (Cache object): { 'float': 3.14159, 'list': [1, 'two', '3', 4], 'str': 'string' }

↪ (<type 'dict'>)

Expectation (Cache object): result = { 'str': 'string', 'float': 3.14159, 'list': [1, 'two',

↪ '3', 4] } (<type 'dict'>)

A.2.9 caching.property_cache_pickle: Test cached data (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /cache_data_test_load_on_init.pkl as cache
 ↵ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↵ 'list']

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ cache_data_test_load_on_init.pkl)

Info Collecting data from cache instance.

PickCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/cache_data_test_load_on_init.pkl)

PickCache: Providing property for 'dict' from cache ({'1': 1, '3': '3', '2': 'two', '4': 4})

PickCache: Providing property for 'unicode' from cache (u'unicode')

PickCache: Providing property for 'str' from cache ('string')

PickCache: Providing property for 'integer' from cache (17)

PickCache: Providing property for 'float' from cache (3.14159)

PickCache: Providing property for 'list' from cache ([1, 'two', '3', 4])

Success Cached data is correct (Content {'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float':
 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3', '2': 'two', '4': 4}, 'str': 'string', 'integer': 17}
 and Type is <type 'dict'>).

Result (Cached data): { 'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float':
 ↵ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '3': '3', '2': 'two', '4': 4 },
 ↵ 'str': 'string', 'integer': 17 } (<type 'dict'>)

Expectation (Cached data): result = { 'uncached': 'uncached_data_of_class', 'dict': { '1': 1,
 ↵ '3': '3', '2': 'two', '4': 4 }, 'unicode': u'unicode', 'str': 'string', 'integer': 17,
 ↵ 'float': 3.14159, 'list': [1, 'two', '3', 4] } (<type 'dict'>)

A.2.10 caching.property_cache_pickle: Test cached data (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /cache_data_test_no_load_on_init.pkl as cache
 ↵ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↵ 'list']

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ cache_data_test_no_load_on_init.pkl)

Info Collecting data from cache instance.

PickCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/cache_data_test_no_load_on_init.pkl)

PickCache: Providing property for 'dict' from cache ({'1': 1, '3': '3', '2': 'two', '4': 4})

PickCache: Providing property for 'unicode' from cache (u'unicode')

PickCache: Providing property for 'str' from cache ('string')

PickCache: Providing property for 'integer' from cache (17)

PickCache: Providing property for 'float' from cache (3.14159)

PickCache: Providing property for 'list' from cache ([1, 'two', '3', 4])

Success Cached data is correct (Content {'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float':
 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '3': '3', '2': 'two', '4': 4}, 'str': 'string', 'integer': 17}
 and Type is <type 'dict'>).

Result (Cached data): { 'uncached': 'uncached_data_of_class', 'unicode': u'unicode', 'float':
 ↵ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '3': '3', '2': 'two', '4': 4 },
 ↵ 'str': 'string', 'integer': 17 } (<type 'dict'>)

Expectation (Cached data): result = { 'uncached': 'uncached_data_of_class', 'dict': { '1': 1,
 ↵ '3': '3', '2': 'two', '4': 4 }, 'unicode': u'unicode', 'str': 'string', 'integer': 17,
 ↵ 'float': 3.14159, 'list': [1, 'two', '3', 4] } (<type 'dict'>)

A.2.11 caching.property_cache_pickle: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

```

Deleting cache file from filesystem to ensure identical conditions for each test run.
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ /data_version_test_load_on_init.pkl as cache
↪ file.
PickCache: Cache file does not exists (yet).
PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
↪ 'list']
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ data_version_test_load_on_init.pkl)

```

Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

```

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↪ output_data/data_version_test_load_on_init.pkl)
PickCache: Data version increased, ignoring previous cache data
PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
↪ 'list']
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ data_version_test_load_on_init.pkl)
PickCache: Providing property for 'dict' from cache ({'1': '1', '3': 'three', '2': 2, '4':
↪ '4'})
PickCache: Providing property for 'unicode' from cache (u'__unicode__')
PickCache: Providing property for 'str' from cache ('__string__')
PickCache: Providing property for 'integer' from cache (34)
PickCache: Providing property for 'float' from cache (2.71828)
PickCache: Providing property for 'list' from cache (['one', 2, 3, '4'])
Result (Instance data after increasing data_version): { 'unicode': u'__unicode__', 'float':
↪ 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4'
↪ }, 'str': '__string__', 'integer': 34 } (<type 'dict'>)
Expectation (Instance data after increasing data_version): result = { 'dict': { '1': '1',
↪ '3': 'three', '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__',
↪ 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ] } (<type 'dict'>)

```

A.2.12 caching.property_cache_pickle: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Unittest for caching

Deleting cache file from filesystem to ensure identical conditions for each test run.

```
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ /data_version_test_no_load_on_init.pkl as cache
↳ file.
```

```
PickCache: Cache file does not exists (yet).
```

```
PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
↳ 'list']
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

Success Instance data after increasing data_version is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

```
PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↳ output_data/data_version_test_no_load_on_init.pkl)
```

```
PickCache: Data version increased, ignoring previous cache data
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'dict' from source instance ({'1': '1', '3': 'three', '2': 2,
↳ '4': '4'})
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'unicode' from source instance (u'__unicode__')
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'str' from source instance ('__string__')
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'integer' from source instance (34)
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'float' from source instance (2.71828)
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
Result (Instance data after increasing data_version): { 'unicode': u'__unicode__', 'float':
↳ 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4'
↳ }, 'str': '__string__', 'integer': 34 } (<type 'dict'>)
```

```
Expectation (Instance data after increasing data_version): result = { 'dict': { '1': '1',
↳ '3': 'three', '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__',
↳ 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ] } (<type 'dict'>)
```


A.2.13 caching.property_cache_pickle: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl
 ↪ as cache file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↪ 'list']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl)

Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '__string__', 'integer': 34} and Type is <type 'dict'>).

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl)

PickCache: Source uid changed, ignoring previous cache data

PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float',
 ↪ 'list']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl)

PickCache: Providing property for 'dict' from cache ({'1': '1', '3': 'three', '2': 2, '4':
 ↪ '4'})

PickCache: Providing property for 'unicode' from cache (u'__unicode__')

PickCache: Providing property for 'str' from cache ('__string__')

PickCache: Providing property for 'integer' from cache (34)

PickCache: Providing property for 'float' from cache (2.71828)

PickCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

Result (Instance data after changing uid): { 'unicode': u'__unicode__', 'float': 2.71828,
 ↪ 'list': ['one', 2, 3, '4'], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4' },
 ↪ 'str': '__string__', 'integer': 34 } (<type 'dict'>)

Expectation (Instance data after changing uid): result = { 'dict': { '1': '1', '3': 'three',
 ↪ '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__', 'integer': 34,
 ↪ 'float': 2.71828, 'list': ['one', 2, 3, '4'] } (<type 'dict'>)

A.2.14 `caching.property_cache_pickle`: Test get from source caused by changed uid (partially init)**Testresult**

This test was passed with the state: **Success**.

Info Initialising `property_cache_pickle` (`load_all_on_init=True`).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with `/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_no_load_on_init.pkl` as cache file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['dict', 'unicode', 'str', 'integer', 'float', 'list']

PickCache: cache-file stored (`/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_no_load_on_init.pkl`)

Success Instance data after changing uid is correct (Content {'unicode': u'__unicode__', 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '3': 'three', '2': 2, '4': '4'}, 'str': '.__string__', 'integer': 34} and Type is <type 'dict'>).

```

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↳ output_data/uid_test_no_load_on_init.pkl)
PickCache: Source uid changed, ignoring previous cache data
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'dict' from source instance ({'1': '1', '3': 'three', '2': 2,
↳ '4': '4'})
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'unicode' from source instance (u'__unicode__')
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'str' from source instance ('__string__')
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'integer' from source instance (34)
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'float' from source instance (2.71828)
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
Result (Instance data after changing uid): { 'unicode': u'__unicode__', 'float': 2.71828,
↳ 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '3': 'three', '2': 2, '4': '4' },
↳ 'str': '__string__', 'integer': 34 } (<type 'dict'>)
Expectation (Instance data after changing uid): result = { 'dict': { '1': '1', '3': 'three',
↳ '2': 2, '4': '4' }, 'unicode': u'__unicode__', 'str': '__string__', 'integer': 34,
↳ 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ] } (<type 'dict'>)

```

A.2.15 caching.property_cache_pickle: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**.

Info Installing save_callback, which sets a variable to True on execution.

Success Save callback execution variable is correct (Content True and Type is <type 'bool'>).

```

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ save_callback_load_on_init.json)
Result (Save callback execution variable): True (<type 'bool'>)
Expectation (Save callback execution variable): result = True (<type 'bool'>)

```

A.2.16 caching.property_cache_pickle: Test internal key usage

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.pkl as
 ↪ cache file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['__property_cache_data_version',
 ↪ '_property_cache_data_version_', '__property_cache_uid_', '_property_cache_uid_']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.pkl)

Info Extracting storage object from property_cache_pickle for comparison.

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.pkl)

Using storage object of cache class for comparison: {'__property_cache_data_version_': u'no

↪ data version', '___property_cache_data_version_': u'no second data version',
 ↪ '__property_cache_uid_': u'no uid', '_property_cache_uid_': 'my-unique-id',
 ↪ '_property_cache_data_version_': 1, '___property_cache_uid_': u'no second uid'}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache is correct (Content {'__property_cache_data_version_': u'no data version',
 '___property_cache_data_version_': u'no second data version', '_property_cache_uid_': u'no uid',
 '___property_cache_uid_': u'no second uid'} and Type is <type 'dict'>).

Result (Cache): { '__property_cache_data_version_': u'no data version',

↪ '___property_cache_data_version_': u'no second data version', '_property_cache_uid_':
 ↪ u'no uid', '___property_cache_uid_': u'no second uid' } (<type 'dict'>)

Expectation (Cache): result = { '__property_cache_data_version_': u'no data version',

↪ '___property_cache_data_version_': u'no second data version', '_property_cache_uid_':
 ↪ u'no uid', '___property_cache_uid_': u'no second uid' } (<type 'dict'>)

Success Keyfilter returnvalue for 5 (<type 'int'>) is correct (Content 5 and Type is <type 'int'>).

Result (Keyfilter returnvalue for 5 (<type 'int'>)): 5 (<type 'int'>)

Expectation (Keyfilter returnvalue for 5 (<type 'int'>)): result = 5 (<type 'int'>)

B Trace for testrun with python 3.8.5 (final)

B.1 Tests with status Info (20)

B.1.1 caching.property_cache.json: Test full initialised JSON-Cache-Object

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.json as cache
↪ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
↪ 'dict']

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.json)

Info Extracting storage object from property_cache.json for comparison.

JsonCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.json)

Using storage object of cache class for comparison: {'_property_cache_data_version': 1,

↪ '_property_cache_uid': 'my_unique_id', 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4},
↪ 'float': 3.14159, 'integer': 17, 'list': [1, 'two', '3', 4], 'str': 'string', 'unicode':
↪ 'unicode'}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache object is correct (Content {'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'float': 3.14159, 'integer': 17, 'list': [1, 'two', '3', 4], 'str': 'string', 'unicode': 'unicode'} and Type is <class 'dict'>).

Result (Cache object): { 'dict': { '1': 1, '2': 'two', '3': '3', '4': 4 }, 'float': 3.14159,

↪ 'integer': 17, 'list': [1, 'two', '3', 4], 'str': 'string', 'unicode': 'unicode' }

↪ (<class 'dict'>)

Expectation (Cache object): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17,

↪ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3',

↪ '4': 4 } } (<class 'dict'>)

B.1.2 caching.property_cache.json: Test partially initialisation of JSON-Cache-Object

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=False).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json as
 ↪ cache file.

JsonCache: Cache file does not exists (yet).

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)

Info Partially initialising cache object by requesting some information.

JsonCache: Loading property for 'str' from source instance ('string')

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)

JsonCache: Loading property for 'unicode' from source instance ('unicode')

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)

JsonCache: Loading property for 'integer' from source instance (17)

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)

Info Extracting storage object from property_cache.json for comparison.

JsonCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.json)

Using storage object of cache class for comparison: {'_property_cache_data_version': 1,

↪ '_property_cache_uid': 'my_unique_id', 'integer': 17, 'str': 'string', 'unicode':

↪ 'unicode'}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache object is correct (Content {'integer': 17, 'str': 'string', 'unicode': 'unicode'} and Type is <class 'dict'>).

Result (Cache object): { 'integer': 17, 'str': 'string', 'unicode': 'unicode' } (<class

↪ 'dict'>)

Expectation (Cache object): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17 }

↪ (<class 'dict'>)

B.1.3 caching.property_cache.json: Test cached data (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /cache_data_test_load_on_init.json as cache
 ↵ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ cache_data_test_load_on_init.json)

Info Collecting data from cache instance.

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/cache_data_test_load_on_init.json)

JsonCache: Providing property for 'str' from cache ('string')

JsonCache: Providing property for 'unicode' from cache ('unicode')

JsonCache: Providing property for 'integer' from cache (17)

JsonCache: Providing property for 'float' from cache (3.14159)

JsonCache: Providing property for 'list' from cache ([1, 'two', '3', 4])

JsonCache: Providing property for 'dict' from cache ({'1': 1, '2': 'two', '3': '3', '4': 4})

JsonCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list':
 [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'} and
 Type is <class 'dict'>).

Result (Cached data): { 'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float':
 ↵ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3', '4': 4 },
 ↵ 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

Expectation (Cached data): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17,
 ↵ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3',
 ↵ '4': 4 }, 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

B.1.4 caching.property_cache.json: Test cached data (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data_1
 ↪ /cache_data_test_no_load_on_init.json as cache
 ↪ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↪ 'dict']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/1
 ↪ cache_data_test_no_load_on_init.json)

Info Collecting data from cache instance.

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/1
 ↪ output_data/cache_data_test_no_load_on_init.json)

JsonCache: Providing property for 'str' from cache ('string')

JsonCache: Providing property for 'unicode' from cache ('unicode')

JsonCache: Providing property for 'integer' from cache (17)

JsonCache: Providing property for 'float' from cache (3.14159)

JsonCache: Providing property for 'list' from cache ([1, 'two', '3', 4])

JsonCache: Providing property for 'dict' from cache ({'1': 1, '2': 'two', '3': '3', '4': 4})

JsonCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'} and Type is <class 'dict'>).

Result (Cached data): { 'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float':
 ↪ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3', '4': 4 },
 ↪ 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

Expectation (Cached data): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17,
 ↪ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3',
 ↪ '4': 4 }, 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

B.1.5 caching.property_cache.json: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /data_version_test_load_on_init.json as cache
 ↵ file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ data_version_test_load_on_init.json)

Success Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__',
 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and
 Type is <class 'dict'>).

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/data_version_test_load_on_init.json)

JsonCache: Data version increased, ignoring previous cache data

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ data_version_test_load_on_init.json)

JsonCache: Providing property for 'str' from cache ('__string__')

JsonCache: Providing property for 'unicode' from cache ('__unicode__')

JsonCache: Providing property for 'integer' from cache (34)

JsonCache: Providing property for 'float' from cache (2.71828)

JsonCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

JsonCache: Providing property for 'dict' from cache ({'1': '1', '2': 2, '3': 'three', '4':
 ↵ '4'})

Result (Instance data after increasing data_version): { 'str': '__string__', 'unicode':
 ↵ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {
 ↵ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

Expectation (Instance data after increasing data_version): result = { 'str': '__string__',
 ↵ 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'],
 ↵ 'dict': { '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

B.1.6 caching.property_cache.json: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

Unittest for caching

```
Deleting cache file from filesystem to ensure identical conditions for each test run.
```

```
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ /data_version_test_no_load_on_init.json as cache  
↪ file.
```

```
JsonCache: Cache file does not exists (yet).
```

```
JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',  
↪ 'dict']
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

Success Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

```
JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/  
↪ output_data/data_version_test_no_load_on_init.json)
```

```
JsonCache: Data version increased, ignoring previous cache data
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'str' from source instance ('__string__')
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'unicode' from source instance ('__unicode__')
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'integer' from source instance (34)
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'float' from source instance (2.71828)
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
JsonCache: Loading property for 'dict' from source instance ({'1': '1', '2': 2, '3': 'three',  
↪ '4': '4'})
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/  
↪ data_version_test_no_load_on_init.json)
```

```
Result (Instance data after increasing data_version): { 'str': '__string__', 'unicode':  
↪ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': {  
↪ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)
```

```
Expectation (Instance data after increasing data_version): result = { 'str': '__string__',  
↪ 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ],  
↪ 'dict': { '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)
```

B.1.7 caching.property_cache_json: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.json
 ↪ as cache file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↪ 'dict']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
 ↪ uid_test_load_on_init.json)

Success Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
 ↪ output_data/uid_test_load_on_init.json)

JsonCache: Source uid changed, ignoring previous cache data

JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↪ 'dict']

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
 ↪ uid_test_load_on_init.json)

JsonCache: Providing property for 'str' from cache ('__string__')

JsonCache: Providing property for 'unicode' from cache ('__unicode__')

JsonCache: Providing property for 'integer' from cache (34)

JsonCache: Providing property for 'float' from cache (2.71828)

JsonCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

JsonCache: Providing property for 'dict' from cache ({'1': '1', '2': 2, '3': 'three', '4':
 ↪ '4'})

Result (Instance data after changing uid): { 'str': '__string__', 'unicode': '__unicode__',
 ↪ 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': { '1': '1', '2':
 ↪ 2, '3': 'three', '4': '4' } } (<class 'dict'>)

Expectation (Instance data after changing uid): result = { 'str': '__string__', 'unicode':
 ↪ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {
 ↪ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

B.1.8 caching.property_cache.json: Test get from source caused by changed uid (partially init)**Testresult**

This test was passed with the state: **Success**.

Info Initialising property_cache.json (load_all_on_init=True).

```
Deleting cache file from filesystem to ensure identical conditions for each test run.
```

```
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data_
↪ /uid_test_no_load_on_init.json as cache
↪ file.
```

```
JsonCache: Cache file does not exists (yet).
```

```
JsonCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
↪ 'dict']
```

```
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
```

Success Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

```

JsonCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↪ output_data/uid_test_no_load_on_init.json)
JsonCache: Source uid changed, ignoring previous cache data
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'str' from source instance ('__string__')
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'unicode' from source instance ('__unicode__')
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'integer' from source instance (34)
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'float' from source instance (2.71828)
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
JsonCache: Loading property for 'dict' from source instance ({'1': '1', '2': 2, '3': 'three',
↪ '4': '4'})
JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ uid_test_no_load_on_init.json)
Result (Instance data after changing uid): { 'str': '__string__', 'unicode': '__unicode__',
↪ 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '2':
↪ 2, '3': 'three', '4': '4' } } (<class 'dict'>)
Expectation (Instance data after changing uid): result = { 'str': '__string__', 'unicode':
↪ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': {
↪ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

```

B.1.9 caching.property_cache.json: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**.

Info Installing save_callback, which sets a variable to True on execution.

Success Save callback execution variable is correct (Content True and Type is <class 'bool'>).

```

JsonCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↪ save_callback_load_on_init.json)
Result (Save callback execution variable): True (<class 'bool'>)
Expectation (Save callback execution variable): result = True (<class 'bool'>)

```

B.1.10 caching.property_cache_json: Test internal key usage

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_json (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.json as
 ↪ cache file.

JsonCache: Cache file does not exists (yet).

JsonCache: Loading all data from source - ['_property_cache_uid_', '__property_cache_uid_',
 ↪ '_property_cache_data_version_', '__property_cache_data_version_']

JsonCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.json)

Info Extracting storage object from property_cache_json for comparison.

JsonCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.json)

Using storage object of cache class for comparison: {'__property_cache_data_version_': 'no
 ↪ second data version', '___property_cache_uid_': 'no second uid',
 ↪ '__property_cache_data_version_': 'no data version', '__property_cache_uid_': 'no uid',
 ↪ '_property_cache_data_version_': 1, '_property_cache_uid_': 'my_unique_id'}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache is correct (Content {'__property_cache_data_version_': 'no second data version',
 '__property_cache_uid_': 'no second uid', '__property_cache_data_version_': 'no data version',
 '__property_cache_uid_': 'no uid'} and Type is <class 'dict'>).

Result (Cache): { '__property_cache_data_version_': 'no second data version',
 ↪ '___property_cache_uid_': 'no second uid', '__property_cache_data_version_': 'no data
 ↪ version', '__property_cache_uid_': 'no uid' } (<class 'dict'>)

Expectation (Cache): result = { '__property_cache_uid_': 'no uid', '___property_cache_uid_':
 ↪ 'no second uid', '__property_cache_data_version_': 'no data version',
 ↪ '___property_cache_data_version_': 'no second data version' } (<class 'dict'>)

Success Keyfilter returnvalue for 5 (<class 'int'>) is correct (Content 5 and Type is <class 'int'>).

Result (Keyfilter returnvalue for 5 (<class 'int'>)): 5 (<class 'int'>)

Expectation (Keyfilter returnvalue for 5 (<class 'int'>)): result = 5 (<class 'int'>)

B.1.11 caching.property_cache_pickle: Test full initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.pkl as cache
 ↪ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↪ 'dict']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.pkl)

Info Extracting storage object from property_cache_pickle for comparison.

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/load_on_init.pkl)

Using storage object of cache class for comparison: {'str': 'string', 'unicode': 'unicode',
 ↪ 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two',
 ↪ '3': '3', '4': 4}, '_property_cache_uid_': 'my_unique_id',
 ↪ '_property_cache_data_version_': 1}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache object is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}} and Type is <class 'dict'>).

Result (Cache object): { 'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float':

↪ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3', '4': 4 } }
 ↪ (<class 'dict'>)

Expectation (Cache object): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17,

↪ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3',
 ↪ '4': 4 } } (<class 'dict'>)

B.1.12 caching.property_cache_pickle: Test partially initialised PICKLE-Cache-Object

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=False).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl as
 ↪ cache file.

PickCache: Cache file does not exists (yet).

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

Info Partially initialising cache object by requesting some information.

PickCache: Loading property for 'str' from source instance ('string')

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

PickCache: Loading property for 'unicode' from source instance ('unicode')

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

PickCache: Loading property for 'integer' from source instance (17)

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

Info Extracting storage object from property_cache_pickle for comparison.

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/no_load_on_init.pkl)

Using storage object of cache class for comparison: {'_property_cache_uid': 'my_unique_id',
 ↪ '_property_cache_data_version': 1, 'str': 'string', 'unicode': 'unicode', 'integer': 17}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache object is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17} and Type is <class 'dict'>).

Result (Cache object): { 'str': 'string', 'unicode': 'unicode', 'integer': 17 } (<class
 ↪ 'dict'>)

Expectation (Cache object): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17 }
 ↪ (<class 'dict'>)

B.1.13 caching.property_cache_pickle: Test cached data (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /cache_data_test_load_on_init.pkl as cache
 ↵ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ cache_data_test_load_on_init.pkl)

Info Collecting data from cache instance.

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/cache_data_test_load_on_init.pkl)

PickCache: Providing property for 'str' from cache ('string')

PickCache: Providing property for 'unicode' from cache ('unicode')

PickCache: Providing property for 'integer' from cache (17)

PickCache: Providing property for 'float' from cache (3.14159)

PickCache: Providing property for 'list' from cache ([1, 'two', '3', 4])

PickCache: Providing property for 'dict' from cache ({'1': 1, '2': 'two', '3': '3', '4': 4})

PickCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list':
 [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'} and
 Type is <class 'dict'>).

Result (Cached data): { 'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float':
 ↵ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3', '4': 4 },
 ↵ 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

Expectation (Cached data): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17,
 ↵ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3',
 ↵ '4': 4 }, 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

B.1.14 caching.property_cache_pickle: Test cached data (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /cache_data_test_no_load_on_init.pkl as cache
 ↵ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ cache_data_test_no_load_on_init.pkl)

Info Collecting data from cache instance.

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/cache_data_test_no_load_on_init.pkl)

PickCache: Providing property for 'str' from cache ('string')

PickCache: Providing property for 'unicode' from cache ('unicode')

PickCache: Providing property for 'integer' from cache (17)

PickCache: Providing property for 'float' from cache (3.14159)

PickCache: Providing property for 'list' from cache ([1, 'two', '3', 4])

PickCache: Providing property for 'dict' from cache ({'1': 1, '2': 'two', '3': '3', '4': 4})

PickCache: Key 'uncached' is not in cached_keys. Uncached data will be returned.

Success Cached data is correct (Content {'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float': 3.14159, 'list':
 [1, 'two', '3', 4], 'dict': {'1': 1, '2': 'two', '3': '3', '4': 4}, 'uncached': 'uncached_data_of_class'} and
 Type is <class 'dict'>).

Result (Cached data): { 'str': 'string', 'unicode': 'unicode', 'integer': 17, 'float':
 ↵ 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3', '4': 4 },
 ↵ 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

Expectation (Cached data): result = { 'str': 'string', 'unicode': 'unicode', 'integer': 17,
 ↵ 'float': 3.14159, 'list': [1, 'two', '3', 4], 'dict': { '1': 1, '2': 'two', '3': '3',
 ↵ '4': 4 }, 'uncached': 'uncached_data_of_class' } (<class 'dict'>)

B.1.15 caching.property_cache_pickle: Test get from source caused by increased data version (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ /data_version_test_load_on_init.pkl as cache
 ↵ file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ data_version_test_load_on_init.pkl)

Success Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__',
 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and
 Type is <class 'dict'>).

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/ ↵
 ↵ output_data/data_version_test_load_on_init.pkl)

PickCache: Data version increased, ignoring previous cache data

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↵ 'dict']

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/ ↵
 ↵ data_version_test_load_on_init.pkl)

PickCache: Providing property for 'str' from cache ('__string__')

PickCache: Providing property for 'unicode' from cache ('__unicode__')

PickCache: Providing property for 'integer' from cache (34)

PickCache: Providing property for 'float' from cache (2.71828)

PickCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

PickCache: Providing property for 'dict' from cache ({'1': '1', '2': 2, '3': 'three', '4':
 ↵ '4'})

Result (Instance data after increasing data_version): { 'str': '__string__', 'unicode':
 ↵ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {
 ↵ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

Expectation (Instance data after increasing data_version): result = { 'str': '__string__',
 ↵ 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'],
 ↵ 'dict': { '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

B.1.16 caching.property_cache_pickle: Test get from source caused by increased data version (partially init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Unittest for caching

Deleting cache file from filesystem to ensure identical conditions for each test run.

```
Initialising cached class with /user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ /data_version_test_no_load_on_init.pkl as cache
↳ file.
```

```
PickCache: Cache file does not exists (yet).
```

```
PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
↳ 'dict']
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

Success Instance data after increasing data_version is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

```
PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↳ output_data/data_version_test_no_load_on_init.pkl)
```

```
PickCache: Data version increased, ignoring previous cache data
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'str' from source instance ('__string__')
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'unicode' from source instance ('__unicode__')
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'integer' from source instance (34)
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'float' from source instance (2.71828)
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
PickCache: Loading property for 'dict' from source instance ({'1': '1', '2': 2, '3': 'three',
↳ '4': '4'})
```

```
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ data_version_test_no_load_on_init.pkl)
```

```
Result (Instance data after increasing data_version): { 'str': '__string__', 'unicode':
↳ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': {
↳ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)
```

```
Expectation (Instance data after increasing data_version): result = { 'str': '__string__',
↳ 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ],
↳ 'dict': { '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)
```

B.1.17 caching.property_cache_pickle: Test get from source caused by changed uid (full init)

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl
 ↪ as cache file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↪ 'dict']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl)

Success Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is <class 'dict'>).

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl)

PickCache: Source uid changed, ignoring previous cache data

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list',
 ↪ 'dict']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_load_on_init.pkl)

PickCache: Providing property for 'str' from cache ('__string__')

PickCache: Providing property for 'unicode' from cache ('__unicode__')

PickCache: Providing property for 'integer' from cache (34)

PickCache: Providing property for 'float' from cache (2.71828)

PickCache: Providing property for 'list' from cache (['one', 2, 3, '4'])

PickCache: Providing property for 'dict' from cache ({'1': '1', '2': 2, '3': 'three', '4':
 ↪ '4'})

Result (Instance data after changing uid): { 'str': '__string__', 'unicode': '__unicode__',
 ↪ 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': { '1': '1', '2':
 ↪ 2, '3': 'three', '4': '4' } } (<class 'dict'>)

Expectation (Instance data after changing uid): result = { 'str': '__string__', 'unicode':
 ↪ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {
 ↪ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

B.1.18 `caching.property_cache_pickle`: Test get from source caused by changed uid (partially init)**Testresult**

This test was passed with the state: **Success**.

Info Initialising `property_cache_pickle` (`load_all_on_init=True`).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with `/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_no_load_on_init.pkl` as cache
↪ `file`.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['str', 'unicode', 'integer', 'float', 'list', 'dict']
↪ 'dict']

PickCache: cache-file stored (`/user_data/data/dirk/prj/unittest/caching/unittest/output_data/uid_test_no_load_on_init.pkl`)

Success Instance data after changing uid is correct (Content {'str': '__string__', 'unicode': '__unicode__', 'integer': 34, 'float': 2.71828, 'list': ['one', 2, 3, '4'], 'dict': {'1': '1', '2': 2, '3': 'three', '4': '4'}} and Type is `<class 'dict'>`).

```

PickCache: Loading properties from cache (/user_data/data/dirk/prj/unittest/caching/unittest/
↳ output_data/uid_test_no_load_on_init.pkl)
PickCache: Source uid changed, ignoring previous cache data
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'str' from source instance ('__string__')
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'unicode' from source instance ('__unicode__')
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'integer' from source instance (34)
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'float' from source instance (2.71828)
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'list' from source instance (['one', 2, 3, '4'])
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
PickCache: Loading property for 'dict' from source instance ({'1': '1', '2': 2, '3': 'three',
↳ '4': '4'})
PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ uid_test_no_load_on_init.pkl)
Result (Instance data after changing uid): { 'str': '__string__', 'unicode': '__unicode__',
↳ 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': { '1': '1', '2':
↳ 2, '3': 'three', '4': '4' } } (<class 'dict'>)
Expectation (Instance data after changing uid): result = { 'str': '__string__', 'unicode':
↳ '__unicode__', 'integer': 34, 'float': 2.71828, 'list': [ 'one', 2, 3, '4' ], 'dict': {
↳ '1': '1', '2': 2, '3': 'three', '4': '4' } } (<class 'dict'>)

```

B.1.19 caching.property_cache_pickle: Test execution of save callback (full init)

Testresult

This test was passed with the state: **Success**.

Info Installing save_callback, which sets a variable to True on execution.

Success Save callback execution variable is correct (Content True and Type is <class 'bool'>).

```

PickCache: cache-file stored (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/
↳ save_callback_load_on_init.json)
Result (Save callback execution variable): True (<class 'bool'>)
Expectation (Save callback execution variable): result = True (<class 'bool'>)

```

B.1.20 caching.property_cache_pickle: Test internal key usage

Testresult

This test was passed with the state: **Success**.

Info Initialising property_cache_pickle (load_all_on_init=True).

Deleting cache file from filesystem to ensure identical conditions for each test run.

Initialising cached class with

↪ /user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.pkl as
 ↪ cache file.

PickCache: Cache file does not exists (yet).

PickCache: Loading all data from source - ['_property_cache_uid_', '__property_cache_uid_',
 ↪ '_property_cache_data_version_', '__property_cache_data_version_']

PickCache: cache-file stored

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.pkl)

Info Extracting storage object from property_cache_pickle for comparison.

PickCache: Loading properties from cache

↪ (/user_data/data/dirk/prj/unittest/caching/unittest/output_data/internal_keys_test.pkl)

Using storage object of cache class for comparison: {'__property_cache_uid_': 'no uid',
 ↪ '___property_cache_uid_': 'no second uid', '__property_cache_data_version_': 'no data
 ↪ version', '___property_cache_data_version_': 'no second data version',
 ↪ '_property_cache_uid_': 'my_unique_id', '_property_cache_data_version_': 1}

Deleting overhead keys: _property_cache_data_version_, _property_cache_uid_

Success Cache is correct (Content {'__property_cache_uid_': 'no uid', '___property_cache_uid_': 'no second uid',
 '_property_cache_data_version_': 'no data version', '___property_cache.data.version_': 'no second data
 version'} and Type is <class 'dict'>).

Result (Cache): { '__property_cache_uid_': 'no uid', '___property_cache_uid_': 'no second
 ↪ uid', '__property_cache_data_version_': 'no data version',
 ↪ '___property_cache_data_version_': 'no second data version' } (<class 'dict'>)

Expectation (Cache): result = { '__property_cache_uid_': 'no uid', '___property_cache_uid_':
 ↪ 'no second uid', '__property_cache_data_version_': 'no data version',
 ↪ '___property_cache_data_version_': 'no second data version' } (<class 'dict'>)

Success Keyfilter returnvalue for 5 (<class 'int'>) is correct (Content 5 and Type is <class 'int'>).

Result (Keyfilter returnvalue for 5 (<class 'int'>)): 5 (<class 'int'>)

Expectation (Keyfilter returnvalue for 5 (<class 'int'>)): result = 5 (<class 'int'>)

C Test-Coverage

C.1 caching

The line coverage for `caching` was 100.0%

The branch coverage for `caching` was 100.0%

C.1.1 `caching.__init__.py`

The line coverage for `caching.__init__.py` was 100.0%

The branch coverage for `caching.__init__.py` was 100.0%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 """
5 caching (Caching Module)
6 =====
7
8 **Author:**
9
10 * Dirk Alders <sudo-dirk@mount-mockery.de>
11
12 **Description:**
13
14     This Module supports functions and classes for caching e.g. properties of other instances.
15
16 **Submodules:**
17
18 * :class:`caching.property_cache_json`
19 * :class:`caching.property_cache_pickle`
20
21 **Unittest:**
22
23     See also the :download:`unittest <../../caching/_testresults_/unittest.pdf>` documentation.
24 """
25 __DEPENDENCIES__ = []
26
27 import hashlib
28 import hmac
29 import json
30 import logging
31 import os
32 import pickle
33 import sys
34
35 try:
36     from config import APP_NAME as ROOT_LOGGER_NAME
37 except ImportError:
38     ROOT_LOGGER_NAME = 'root'
39 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
40
41 __DESCRIPTION__ = """The Module {\tt %s} is designed to store information in {\tt json} or {\tt pickle} files to support them much faster then generating them from the original source file.
42 For more Information read the documentation.""" % __name__.replace('-', '\\-')
43 """ The Module Description """
44 __INTERPRETER__ = (2, 3)

```

Unittest for caching

```
45 """The Tested Interpreter—Versions"""
46
47
48 class property_cache_pickle(object):
49     """
50     Class to cache properties, which take longer on initialising than reading a file in pickle
51     format.
52
53     :param source_instance: The source instance holding the data
54     :type source_instance: instance
55     :param cache_filename: File name, where the properties are stored as cache
56     :type cache_filename: str
57     :param load_all_on_init: Optionally init behaviour control parameter. True will load all
58     available properties from source on init, False not.
59     :raises: ?
60
61     .. note:: This module uses logging. So logging should be initialised at least by executing
62     logging.basicConfig(...)
63
64     .. note:: source_instance needs to have at least the following methods: uid(), keys(),
65     data_version(), get()
66
67         * uid(): returns the unique id of the source.
68         * keys(): returns a list of all available keys.
69         * data_version(): returns a version number of the current data (it should be
70     increased, if the get method of the source instance returns improved values or the data
71     structure had been changed).
72         * get(key, default): returns the property for a key. If key does not exists,
73     default will be returned.
74
75     Reasons for updating the complete data set:
76
77     * UID of source_instance has changed (in comparison to the cached value).
78     * data_version is increased
79
80     **Example:**
81
82     .. literalinclude:: ../../caching/_examples_/property_cache_pickle.py
83
84     Will result on the first execution to the following output (with a long execution time):
85
86     .. literalinclude:: ../../caching/_examples_/property_cache_pickle-1.log
87
88     With every following execution (slow for getting "two" which is not cached — see
89     implementation):
90
91     .. literalinclude:: ../../caching/_examples_/property_cache_pickle-2.log
92     """
93     LOG_PREFIX = 'PickCache:'
94     DATA_VERSION_TAG = '_property_cache_data_version_'
95     UID_TAG = '_property_cache_uid_'
96
97     def __init__(self, source_instance, cache_filename, load_all_on_init=False,
98                 callback_on_data_storage=None):
99         self._source_instance = source_instance
100        self._cache_filename = cache_filename
101        self._load_all_on_init = load_all_on_init
102        self._callback_on_data_storage = callback_on_data_storage
103        self._cached_props = None
104
105    def get(self, key, default=None):
```

Unittest for caching

```
97     """
98     Method to get the cached property. If key does not exists in cache, the property will be
    loaded from source_instance and stored in cache (file).
99
100     :param key: key for value to get.
101     :param default: value to be returned, if key does not exists.
102     :returns: value for a given key or default value.
103     """
104     if key in self.keys():
105         if self._cached_props is None:
106             self._init_cache()
107         if self._key_filter(key) not in self._cached_props:
108             val = self._source_instance.get(key, None)
109             logger.debug("%s Loading property for '%s' from source instance (%s)", self.
LOG_PREFIX, key, repr(val))
110             self._cached_props[self._key_filter(key)] = val
111             self._save_cache()
112         else:
113             logger.debug("%s Providing property for '%s' from cache (%s)", self.LOG_PREFIX,
key, repr(self._cached_props.get(self._key_filter(key), default)))
114             return self._cached_props.get(self._key_filter(key), default)
115         else:
116             logger.info("%s Key '%s' is not in cached_keys. Uncached data will be returned.",
self.LOG_PREFIX, key)
117             return self._source_instance.get(key, default)
118
119     def keys(self):
120         return self._source_instance.keys()
121
122     def _data_version(self):
123         if self._cached_props is None:
124             return None
125         else:
126             return self._cached_props.get(self.DATA_VERSION_TAG, None)
127
128     def _init_cache(self):
129         if not self._load_cache() or self._source_instance.uid() != self._uid() or self.
_source_instance.data_version() > self._data_version():
130             if self._uid() is not None and self._source_instance.uid() != self._uid():
131                 logger.debug("%s Source uid changed, ignoring previous cache data", self.
LOG_PREFIX)
132             if self._data_version() is not None and self._source_instance.data_version() > self.
_data_version():
133                 logger.debug("%s Data version increased, ignoring previous cache data", self.
LOG_PREFIX)
134             self._cached_props = dict()
135             if self._load_all_on_init:
136                 self._load_source()
137             self._cached_props[self.UID_TAG] = self._source_instance.uid()
138             self._cached_props[self.DATA_VERSION_TAG] = self._source_instance.data_version()
139             self._save_cache()
140
141     def _load_cache(self):
142         if os.path.exists(self._cache_filename):
143             with open(self._cache_filename, 'rb') as fh:
144                 self._cached_props = pickle.load(fh)
145             logger.info('%s Loading properties from cache (%s)', self.LOG_PREFIX, self.
_cache_filename)
146             return True
147         else:
148             logger.debug('%s Cache file does not exists (yet).', self.LOG_PREFIX)
149             return False
```

Unittest for caching

```
150
151 def _key_filter(self, key):
152     if sys.version_info >= (3, 0):
153         tps = [str]
154     else:
155         tps = [str, unicode]
156     if type(key) in tps:
157         if key.endswith(self.DATA_VERSION_TAG) or key.endswith(self.UID_TAG):
158             return '_' + key
159     return key
160
161 def _load_source(self):
162     logger.debug('%s Loading all data from source - %s', self.LOG_PREFIX, repr(self._source_instance.keys()))
163     for key in self._source_instance.keys():
164         val = self._source_instance.get(key)
165         self._cached_props[self._key_filter(key)] = val
166
167 def _save_cache(self):
168     with open(self._cache_filename, 'wb') as fh:
169         pickle.dump(self._cached_props, fh)
170         logger.info('%s cache-file stored (%s)', self.LOG_PREFIX, self._cache_filename)
171     if self._callback_on_data_storage is not None:
172         self._callback_on_data_storage()
173
174 def _uid(self):
175     if self._cached_props is None:
176         return None
177     else:
178         return self._cached_props.get(self.UID_TAG, None)
179
180
181 class property_cache_json(property_cache_pickle):
182     """
183     Class to cache properties, which take longer on initialising than reading a file in json
184     format. See also parent :py:class:`property_cache_pickle`
185
186     :param source_instance: The source instance holding the data
187     :type source_instance: instance
188     :param cache_filename: File name, where the properties are stored as cache
189     :type cache_filename: str
190     :param load_all_on_init: Optionally init behaviour control parameter. True will load all
191     available properties from source on init, False not.
192     :raises: ?
193
194     .. warning::
195         * This class uses json. You should only use keys of type string!
196         * Unicode types are transfered to strings
197
198     .. note:: This module uses logging. So logging should be initialised at least by executing
199     logging.basicConfig(...)
200
201     .. note:: source_instance needs to have at least the following methods: uid(), keys(),
202     data_version(), get()
203
204         * uid(): returns the unique id of the source.
205         * keys(): returns a list of all available keys.
206         * data_version(): returns a version number of the current data (it should be
207     increased, if the get method of the source instance returns improved values or the data
208     structure had been changed).
209         * get(key, default): returns the property for a key. If key does not exists,
210     default will be returned.
```

Unittest for caching

```
204
205 Reasons for updating the complete data set:
206
207 * UID of source_instance has changed (in comparison to the cached value).
208 * data_version is increased
209
210 **Example:**
211
212 .. literalinclude:: ../../caching/_examples_/property_cache_json.py
213
214 Will result on the first execution to the following output (with a long execution time):
215
216 .. literalinclude:: ../../caching/_examples_/property_cache_json_1.log
217
218 With every following execution (slow for getting "two" which is not cached – see
219 implementation):
220
221 .. literalinclude:: ../../caching/_examples_/property_cache_json_2.log
222 """
223 LOG_PREFIX = 'JsonCache: '
224
225 def _load_cache(self):
226     if os.path.exists(self._cache_filename):
227         with open(self._cache_filename, 'r') as fh:
228             self._cached_props = json.load(fh)
229             logger.info('%s Loading properties from cache (%s)', self.LOG_PREFIX, self._cache_filename)
230         return True
231     else:
232         logger.debug('%s Cache file does not exists (yet).', self.LOG_PREFIX)
233     return False
234
235 def _save_cache(self):
236     with open(self._cache_filename, 'w') as fh:
237         json.dump(self._cached_props, fh, sort_keys=True, indent=4)
238         logger.info('%s cache-file stored (%s)', self.LOG_PREFIX, self._cache_filename)
239     if self._callback_on_data_storage is not None:
240         self._callback_on_data_storage()
```