

## RAPP Platform Tests - Knowrob Wrapper

Generated by Doxygen 1.8.6

Thu Dec 3 2015 15:01:03



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Namespace List . . . . .	1
<b>2</b>	<b>Hierarchical Index</b>	<b>3</b>
2.1	Class Hierarchy . . . . .	3
<b>3</b>	<b>Class Index</b>	<b>5</b>
3.1	Class List . . . . .	5
<b>4</b>	<b>File Index</b>	<b>7</b>
4.1	File List . . . . .	7
<b>5</b>	<b>Namespace Documentation</b>	<b>9</b>
5.1	cognitive_exercise_system_knowrob_services_functional_tests Namespace Reference . . . . .	9
5.1.1	Variable Documentation . . . . .	9
5.1.1.1	PKG . . . . .	9
5.2	load_dump_ontology_functional_tests Namespace Reference . . . . .	9
5.2.1	Variable Documentation . . . . .	9
5.2.1.1	PKG . . . . .	9
5.3	sub_super_class_functional_tests Namespace Reference . . . . .	10
5.3.1	Variable Documentation . . . . .	10
5.3.1.1	PKG . . . . .	10
<b>6</b>	<b>Class Documentation</b>	<b>11</b>
6.1	cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc Class Reference . .	11
6.1.1	Detailed Description . . . . .	12
6.1.2	Member Function Documentation . . . . .	12
6.1.2.1	test_cognitive_tests_of_existent_type . . . . .	12
6.1.2.2	test_cognitive_tests_of_nonexistent_type . . . . .	13
6.1.2.3	test_create_nonexistent_cognitive_test . . . . .	13
6.1.2.4	test_create_nonexistent_cognitive_test_xml_path_nonexistent . . . . .	13
6.1.2.5	test_record_user_performance_nonexisting_test_and_user . . . . .	13
6.1.2.6	test_user_performance_existent_user_and_test . . . . .	13

6.1.2.7	test_user_performance_nonexistent_user_and_test	13
6.2	load_dump_ontology_functional_tests.OntologyFunc Class Reference	14
6.2.1	Detailed Description	14
6.2.2	Member Function Documentation	15
6.2.2.1	test_dump_correct_path	15
6.2.2.2	test_dump_invalid_path	15
6.2.2.3	test_load_existent	15
6.2.2.4	test_load_nonexistent	15
6.3	sub_super_class_functional_tests.OntologyFunc Class Reference	15
6.3.1	Detailed Description	17
6.3.2	Member Function Documentation	17
6.3.2.1	test_sub_superclasses_of_existent_class_direct	17
6.3.2.2	test_sub_superclasses_of_existent_class_direct_recursive	17
6.3.2.3	test_sub_superclasses_of_existent_class_indirect	17
6.3.2.4	test_sub_superclasses_of_existent_class_indirect_recursive	17
6.3.2.5	test_sub_superclasses_of_existent_class_recursive	17
6.3.2.6	test_sub_superclasses_of_existent_classes_recursive	17
6.3.2.7	test_sub_superclasses_of_non_existent_class	17
6.3.2.8	test_sub_superclasses_of_non_existent_classes	17
6.3.2.9	test_subclasses_of_existent_class	18
6.3.2.10	test_subclasses_of_existent_class_recursive	18
6.3.2.11	test_subclasses_of_existent_class_stress	18
6.3.2.12	test_subclasses_of_non_existent_class	18
6.3.2.13	test_subclasses_of_non_existent_class_recursive	18
6.3.2.14	test_superclasses_of_existent_class	18
6.3.2.15	test_superclasses_of_existent_class_recursive	18
6.3.2.16	test_superclasses_of_non_existent_class	18
6.3.2.17	test_superclasses_of_non_existent_class_recursive	18
<b>7</b>	<b>File Documentation</b>	<b>21</b>
7.1	/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/cognitive_exercise_system_knowrob_services_functional_tests.py File Reference	21
7.2	/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/load_dump_ontology_functional_tests.py File Reference	21
7.3	/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/sub_super_class_functional_tests.py File Reference	22
<b>Index</b>		<b>23</b>

# Chapter 1

## Namespace Index

### 1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">cognitive_exercise_system_knowrob_services_functional_tests</a> . . . . .	9
<a href="#">load_dump_ontology_functional_tests</a> . . . . .	9
<a href="#">sub_super_class_functional_tests</a> . . . . .	10



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

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

TestCase	
cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc . . . . .	<a href="#">11</a>
load_dump_ontology_functional_tests.OntologyFunc . . . . .	<a href="#">14</a>
sub_super_class_functional_tests.OntologyFunc . . . . .	<a href="#">15</a>





## Chapter 3

# Class Index

### 3.1 Class List

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

<a href="#">cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc</a>	
Inherits the unittest.TestCase class in order to offer functional tests functionality . . . . .	11
<a href="#">load_dump_ontology_functional_tests.OntologyFunc</a>	
Inherits the unittest.TestCase class in order to offer functional tests functionalit . . . . .	14
<a href="#">sub_super_class_functional_tests.OntologyFunc</a>	
Inherits the unittest.TestCase class in order to offer functional tests functionality . . . . .	15



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/ <a href="#">cognitive_exercise_- system_knowrob_services_functional_tests.py</a> . . . . .	21
/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/ <a href="#">load_dump_ontology_- functional_tests.py</a> . . . . .	21
/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/ <a href="#">sub_super_class_functional- _tests.py</a> . . . . .	22



## Chapter 5

# Namespace Documentation

### 5.1 cognitive\_exercise\_system\_knowrob\_services\_functional\_tests Namespace Reference

#### Classes

- class [OntologyFunc](#)  
*Inherits the unittest.TestCase class in order to offer functional tests functionality.*

#### Variables

- string [PKG](#) = 'rapp\_knowrob\_wrapper'

#### 5.1.1 Variable Documentation

5.1.1.1 string `cognitive_exercise_system_knowrob_services_functional_tests.PKG` = 'rapp\_knowrob\_wrapper'

Definition at line 17 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

### 5.2 load\_dump\_ontology\_functional\_tests Namespace Reference

#### Classes

- class [OntologyFunc](#)  
*Inherits the unittest.TestCase class in order to offer functional tests functionalit.*

#### Variables

- string [PKG](#) = 'rapp\_knowrob\_wrapper'

#### 5.2.1 Variable Documentation

5.2.1.1 string `load_dump_ontology_functional_tests.PKG` = 'rapp\_knowrob\_wrapper'

Definition at line 18 of file `load_dump_ontology_functional_tests.py`.

## 5.3 sub\_super\_class\_functional\_tests Namespace Reference

### Classes

- class [OntologyFunc](#)  
*Inherits the `unittest.TestCase` class in order to offer functional tests functionality.*

### Variables

- string [PKG](#) = 'rapp\_knowrob\_wrapper'

#### 5.3.1 Variable Documentation

##### 5.3.1.1 string sub\_super\_class\_functional\_tests.PKG = 'rapp\_knowrob\_wrapper'

Definition at line 17 of file sub\_super\_class\_functional\_tests.py.

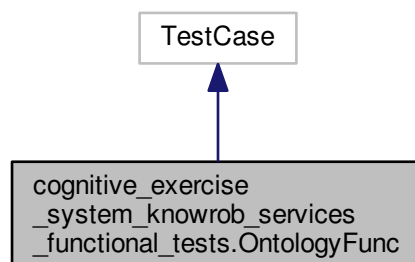
## Chapter 6

# Class Documentation

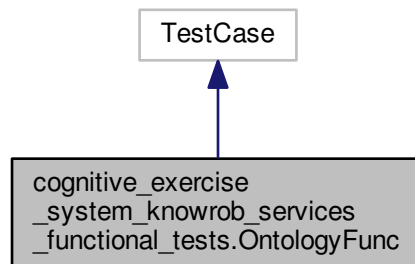
### 6.1 `cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc` Class Reference

Inherits the `unittest.TestCase` class in order to offer functional tests functionality.

Inheritance diagram for `cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc`:



Collaboration diagram for `cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc`:



## Public Member Functions

- def [test\\_cognitive\\_tests\\_of\\_existent\\_type](#)  
*Test cognitive tests of existent type.*
- def [test\\_cognitive\\_tests\\_of\\_nonexistent\\_type](#)  
*Test cognitive tests of non existent type.*
- def [test\\_create\\_nonexistent\\_cognitive\\_test](#)  
*Test create existent cognitive test* def `test_create_existent_cognitive_test(self): subclasses_of_service = rospy.get_param(\ "rapp_knowrob_wrapper_create_cognitive_tests") rospy.wait_for_service(subclasses_of_service)`
- def [test\\_create\\_nonexistent\\_cognitive\\_test\\_xml\\_path\\_nonexistent](#)  
*Test create non existent cognitive test with invalid xml path.*
- def [test\\_record\\_user\\_performance\\_nonexisting\\_test\\_and\\_user](#)  
*Test record user performance of existent user and test type* def `test_record_user_performance_existing_test_and_user(self): subclasses_of_service = rospy.get_param(\ "rapp_knowrob_wrapper_record_user_cognitive_tests_performance") rospy.wait_for_service(subclasses_of_service)`
- def [test\\_user\\_performance\\_existent\\_user\\_and\\_test](#)  
*Test user performance of existent user and test type.*
- def [test\\_user\\_performance\\_nonexistent\\_user\\_and\\_test](#)  
*Test user performance of nonexistent user and test type.*

### 6.1.1 Detailed Description

Inherits the `unittest.TestCase` class in order to offer functional tests functionality.

Definition at line 41 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

### 6.1.2 Member Function Documentation

#### 6.1.2.1 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_cognitive_tests_of_existent_type ( self )`

Test cognitive tests of existent type.

Definition at line 44 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.



6.1.2.2 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_cognitive_tests_of_nonexistent_type ( self )`

Test cognitive tests of non existent type.

Definition at line 62 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

6.1.2.3 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_create_nonexistent_cognitive_test ( self )`

Test create existent cognitive test `def test_create_existent_cognitive_test(self): subclasses_of_service = rospy.get_param(\ "rapp_knowrob_wrapper_create_cognitive_tests") rospy.wait_for_service(subclasses_of_service)`

Test create non existent cognitive test

Definition at line 166 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

6.1.2.4 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_create_nonexistent_cognitive_test_xml_path_nonexistent ( self )`

Test create non existent cognitive test with invalid xml path.

Definition at line 186 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

6.1.2.5 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_record_user_performance_nonexisting_test_and_user ( self )`

Test record user performance of existent user and test type `def test_record_user_performance_existent_test_and_user(self): subclasses_of_service = rospy.get_param(\ "rapp_knowrob_wrapper_record_user_cognitive_tests_performance") rospy.wait_for_service(subclasses_of_service)`

`req.test_type="ArithmeticCts" req.test="ArithmeticCts_MewmmEsP" req.score=11 req.timestamp=1 response = test_service(req) self.assertEqual(response.success, True)` Test record user performance of nonexistent user and test type

Definition at line 131 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

6.1.2.6 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_user_performance_existent_user_and_test ( self )`

Test user performance of existent user and test type.

Definition at line 78 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

6.1.2.7 `def cognitive_exercise_system_knowrob_services_functional_tests.OntologyFunc.test_user_performance_nonexistent_user_and_test ( self )`

Test user performance of nonexistent user and test type.

Definition at line 97 of file `cognitive_exercise_system_knowrob_services_functional_tests.py`.

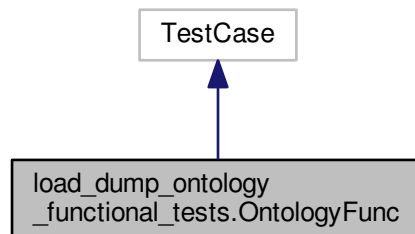
The documentation for this class was generated from the following file:

- [/home/travis/rapp\\_temp/rapp-platform/rapp\\_knowrob\\_wrapper/tests/functional/cognitive\\_exercise\\_system\\_knowrob\\_services\\_functional\\_tests.py](/home/travis/rapp_temp/rapp-platform/rapp_knowrob_wrapper/tests/functional/cognitive_exercise_system_knowrob_services_functional_tests.py)

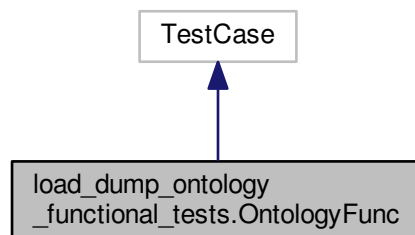
## 6.2 load\_dump\_ontology\_functional\_tests.OntologyFunc Class Reference

Inherits the unittest.TestCase class in order to offer functional tests functionalit.

Inheritance diagram for load\_dump\_ontology\_functional\_tests.OntologyFunc:



Collaboration diagram for load\_dump\_ontology\_functional\_tests.OntologyFunc:



### Public Member Functions

- def [test\\_dump\\_correct\\_path](#)  
*Test dump ontology with correct path.*
- def [test\\_dump\\_invalid\\_path](#)  
*Test dump ontology with invalid path.*
- def [test\\_load\\_existent](#)  
*Test load ontology from existent file.*
- def [test\\_load\\_nonexistent](#)  
*Test load ontology from non existent file.*

### 6.2.1 Detailed Description

Inherits the unittest.TestCase class in order to offer functional tests functionalit.

Definition at line 33 of file `load_dump_ontology_functional_tests.py`.

## 6.2.2 Member Function Documentation

### 6.2.2.1 def load\_dump\_ontology\_functional\_tests.OntologyFunc.test\_dump\_correct\_path ( self )

Test dump ontology with correct path.

Definition at line 64 of file load\_dump\_ontology\_functional\_tests.py.

### 6.2.2.2 def load\_dump\_ontology\_functional\_tests.OntologyFunc.test\_dump\_invalid\_path ( self )

Test dump ontology with invalid path.

Definition at line 78 of file load\_dump\_ontology\_functional\_tests.py.

### 6.2.2.3 def load\_dump\_ontology\_functional\_tests.OntologyFunc.test\_load\_existent ( self )

Test load ontology from existent file.

Definition at line 36 of file load\_dump\_ontology\_functional\_tests.py.

### 6.2.2.4 def load\_dump\_ontology\_functional\_tests.OntologyFunc.test\_load\_nonexistent ( self )

Test load ontology from non existent file.

Definition at line 50 of file load\_dump\_ontology\_functional\_tests.py.

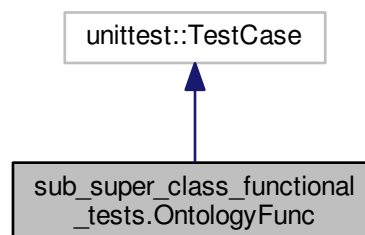
The documentation for this class was generated from the following file:

- [/home/travis/rapp\\_temp/rapp-platform/rapp\\_knowrob\\_wrapper/tests/functional/load\\_dump\\_ontology\\_functional\\_tests.py](#)

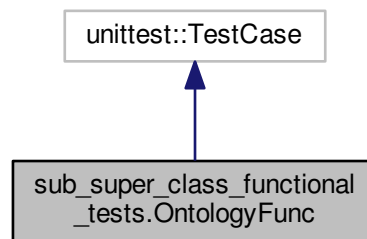
## 6.3 sub\_super\_class\_functional\_tests.OntologyFunc Class Reference

Inherits the unittest.TestCase class in order to offer functional tests functionality.

Inheritance diagram for sub\_super\_class\_functional\_tests.OntologyFunc:



Collaboration diagram for `sub_super_class_functional_tests.OntologyFunc`:



## Public Member Functions

- def [test\\_sub\\_superclasses\\_of\\_existent\\_class\\_direct](#)  
*Tests sub-superclasses\_of existent class direct.*
- def [test\\_sub\\_superclasses\\_of\\_existent\\_class\\_direct\\_recursive](#)  
*Tests sub-superclasses\_of existent class direct recursive.*
- def [test\\_sub\\_superclasses\\_of\\_existent\\_class\\_indirect](#)  
*Tests sub-superclasses\_of existent class indirect.*
- def [test\\_sub\\_superclasses\\_of\\_existent\\_class\\_indirect\\_recursive](#)  
*Tests sub-superclasses\_of existent class indirect recursive.*
- def [test\\_sub\\_superclasses\\_of\\_existent\\_class\\_recursive](#)  
*Tests sub-superclasses\_of existent class recursive.*
- def [test\\_sub\\_superclasses\\_of\\_existent\\_classes\\_recursive](#)  
*Tests sub-superclasses\_of existent classes.*
- def [test\\_sub\\_superclasses\\_of\\_non\\_existent\\_class](#)  
*Tests sub-superclasses\_of non existent class.*
- def [test\\_sub\\_superclasses\\_of\\_non\\_existent\\_classes](#)  
*Tests sub-superclasses\_of non existent classes.*
- def [test\\_subclasses\\_of\\_existent\\_class](#)  
*Tests Subclasses\_of existent class.*
- def [test\\_subclasses\\_of\\_existent\\_class\\_recursive](#)  
*Tests Subclass\_of existent class recursive.*
- def [test\\_subclasses\\_of\\_existent\\_class\\_stress](#)  
*Tests Subclasses\_of existent class stress.*
- def [test\\_subclasses\\_of\\_non\\_existent\\_class](#)  
*Tests Subclass\_of non existent class.*
- def [test\\_subclasses\\_of\\_non\\_existent\\_class\\_recursive](#)  
*Tests Subclass\_of non existent class recursive.*
- def [test\\_superclasses\\_of\\_existent\\_class](#)  
*Tests Superclasses\_of existent class.*
- def [test\\_superclasses\\_of\\_existent\\_class\\_recursive](#)  
*Tests Superclasses\_of existent class recursive.*
- def [test\\_superclasses\\_of\\_non\\_existent\\_class](#)  
*Tests Superclasses\_of non existent class.*
- def [test\\_superclasses\\_of\\_non\\_existent\\_class\\_recursive](#)  
*Tests Superclasses\_of non existent class recursive.*

### 6.3.1 Detailed Description

Inherits the unittest.TestCase class in order to offer functional tests functionality.

Definition at line 38 of file sub\_super\_class\_functional\_tests.py.

### 6.3.2 Member Function Documentation

#### 6.3.2.1 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_existent\_class\_direct ( self )

Tests sub-superclasses\_of existent class direct.

Definition at line 200 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.2 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_existent\_class\_direct\_recursive ( self )

Tests sub-superclasses\_of existent class direct recursive.

Definition at line 234 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.3 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_existent\_class\_indirect ( self )

Tests sub-superclasses\_of existent class indirect.

Definition at line 217 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.4 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_existent\_class\_indirect\_recursive ( self )

Tests sub-superclasses\_of existent class indirect recursive.

Definition at line 251 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.5 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_existent\_class\_recursive ( self )

Tests sub-superclasses\_of existent class recursive.

Definition at line 287 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.6 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_existent\_classes\_recursive ( self )

Tests sub-superclasses\_of existent classes.

Definition at line 325 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.7 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_non\_existent\_class ( self )

Tests sub-superclasses\_of non existent class.

Definition at line 268 of file sub\_super\_class\_functional\_tests.py.

#### 6.3.2.8 def sub\_super\_class\_functional\_tests.OntologyFunc.test\_sub\_superclasses\_of\_non\_existent\_classes ( self )

Tests sub-superclasses\_of non existent classes.

Definition at line 306 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.9** `def sub_super_class_functional_tests.OntologyFunc.test_subclasses_of_existent_class ( self )`

Tests Subclasses\_of existent class.

Definition at line 41 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.10** `def sub_super_class_functional_tests.OntologyFunc.test_subclasses_of_existent_class_recursive ( self )`

Tests Subclass\_of existent class recursive.

Definition at line 92 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.11** `def sub_super_class_functional_tests.OntologyFunc.test_subclasses_of_existent_class_stress ( self )`

Tests Subclasses\_of existent class stress.

Definition at line 58 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.12** `def sub_super_class_functional_tests.OntologyFunc.test_subclasses_of_non_existent_class ( self )`

Tests Subclass\_of non existent class.

Definition at line 75 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.13** `def sub_super_class_functional_tests.OntologyFunc.test_subclasses_of_non_existent_class_recursive ( self )`

Tests Subclass\_of non existent class recursive.

Definition at line 113 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.14** `def sub_super_class_functional_tests.OntologyFunc.test_superclasses_of_existent_class ( self )`

Tests Superclasses\_of existent class.

Definition at line 131 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.15** `def sub_super_class_functional_tests.OntologyFunc.test_superclasses_of_existent_class_recursive ( self )`

Tests Superclasses\_of existent class recursive.

Definition at line 146 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.16** `def sub_super_class_functional_tests.OntologyFunc.test_superclasses_of_non_existent_class ( self )`

Tests Superclasses\_of non existent class.

Definition at line 167 of file sub\_super\_class\_functional\_tests.py.

**6.3.2.17** `def sub_super_class_functional_tests.OntologyFunc.test_superclasses_of_non_existent_class_recursive ( self )`

Tests Superclasses\_of non existent class recursive.

Definition at line 183 of file sub\_super\_class\_functional\_tests.py.

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

- [/home/travis/rapp\\_temp/rapp-platform/rapp\\_knowrob\\_wrapper/tests/functional/sub\\_super\\_class\\_functional-\\_tests.py](#)





## Chapter 7

# File Documentation

### 7.1 /home/travis/rapp\_temp/rapp-platform/rapp-knowrob\_wrapper/tests/functional/cognitive-exercise\_system-knowrob\_services\_functional\_tests.py File Reference

#### Classes

- class [cognitive\\_exercise\\_system-knowrob\\_services\\_functional\\_tests.OntologyFunc](#)  
*Inherits the unittest.TestCase class in order to offer functional tests functionality.*

#### Namespaces

- [cognitive\\_exercise\\_system-knowrob\\_services\\_functional\\_tests](#)

#### Variables

- string [cognitive\\_exercise\\_system-knowrob\\_services\\_functional\\_tests.PKG](#) = 'rapp-knowrob\_wrapper'

### 7.2 /home/travis/rapp\_temp/rapp-platform/rapp-knowrob\_wrapper/tests/functional/load-dump\_ontology\_functional\_tests.py File Reference

#### Classes

- class [load\\_dump\\_ontology\\_functional\\_tests.OntologyFunc](#)  
*Inherits the unittest.TestCase class in order to offer functional tests functionalit.*

#### Namespaces

- [load\\_dump\\_ontology\\_functional\\_tests](#)

#### Variables

- string [load\\_dump\\_ontology\\_functional\\_tests.PKG](#) = 'rapp-knowrob\_wrapper'

### 7.3 /home/travis/rapp\_temp/rapp-platform/rapp\_knowrob\_wrapper/tests/functional/sub\_super\_class\_functional\_tests.py File Reference

#### Classes

- class [sub\\_super\\_class\\_functional\\_tests.OntologyFunc](#)  
*Inherits the `unittest.TestCase` class in order to offer functional tests functionality.*

#### Namespaces

- [sub\\_super\\_class\\_functional\\_tests](#)

#### Variables

- string [sub\\_super\\_class\\_functional\\_tests.PKG](#) = 'rapp\_knowrob\_wrapper'

# Index

`/home/travis/rapp_temp/rapp-platform/rapp_knowrob_`  
    `_wrapper/tests/functional/load_dump_`  
    `ontology_functional_tests.py`, [21](#)

`/home/travis/rapp_temp/rapp-platform/rapp_knowrob_`  
    `wrapper/tests/functional/sub_super_class_`  
    `functional_tests.py`, [22](#)

`cognitive_exercise_system_knowrob_services_functional_`  
    `_tests`, [9](#)

    PKG, [9](#)

`cognitive_exercise_system_knowrob_services_functional_`  
    `_tests.OntologyFunc`, [11](#)

`load_dump_ontology_functional_tests`, [9](#)

    PKG, [9](#)

`load_dump_ontology_functional_tests.OntologyFunc`,  
    [14](#)

`load_dump_ontology_functional_tests::OntologyFunc`  
    `test_dump_correct_path`, [15](#)  
    `test_dump_invalid_path`, [15](#)  
    `test_load_existent`, [15](#)  
    `test_load_nonexistent`, [15](#)

PKG

`cognitive_exercise_system_knowrob_services_`  
    `functional_tests`, [9](#)

`load_dump_ontology_functional_tests`, [9](#)

`sub_super_class_functional_tests`, [10](#)

`sub_super_class_functional_tests`, [10](#)

    PKG, [10](#)

`sub_super_class_functional_tests.OntologyFunc`, [15](#)

`test_cognitive_tests_of_existent_type`  
    `cognitive_exercise_system_knowrob_services_`  
    `functional_tests::OntologyFunc`, [12](#)

`test_create_nonexistent_cognitive_test`  
    `cognitive_exercise_system_knowrob_services_`  
    `functional_tests::OntologyFunc`, [13](#)

`test_dump_correct_path`  
    `load_dump_ontology_functional_tests::Ontology-`  
    `Func`, [15](#)

`test_dump_invalid_path`  
    `load_dump_ontology_functional_tests::Ontology-`  
    `Func`, [15](#)

`test_load_existent`  
    `load_dump_ontology_functional_tests::Ontology-`  
    `Func`, [15](#)

`test_load_nonexistent`  
    `load_dump_ontology_functional_tests::Ontology-`  
    `Func`, [15](#)

`test_sub_superclasses_of_existent_class_direct`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_sub_superclasses_of_existent_class_indirect`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_sub_superclasses_of_existent_class_recursive`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_sub_superclasses_of_existent_classes_recursive`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_sub_superclasses_of_non_existent_class`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_sub_superclasses_of_non_existent_classes`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_subclasses_of_existent_class`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [17](#)

`test_subclasses_of_existent_class_recursive`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_subclasses_of_existent_class_stress`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_subclasses_of_non_existent_class`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_subclasses_of_non_existent_class_recursive`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_superclasses_of_existent_class`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_superclasses_of_existent_class_recursive`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_superclasses_of_non_existent_class`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)

`test_superclasses_of_non_existent_class_recursive`  
    `sub_super_class_functional_tests::OntologyFunc`,  
    [18](#)