

cxcpamx-2.0  
2.0.2\_01

Generated by Doxygen 1.8.15



---

<b>1 cxcparamx</b>	<b>1</b>
1.1 Introduction	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 Class Documentation</b>	<b>5</b>
3.1 cxcparamx::simple Class Reference	5
3.1.1 Detailed Description	6
3.1.2 Constructor & Destructor Documentation	6
3.1.2.1 simple() [1/2]	6
3.1.2.2 simple() [2/2]	7
3.1.3 Member Function Documentation	7
3.1.3.1 get() [1/7]	7
3.1.3.2 get() [2/7]	7
3.1.3.3 get() [3/7]	8
3.1.3.4 get() [4/7]	8
3.1.3.5 get() [5/7]	9
3.1.3.6 get() [6/7]	9
3.1.3.7 get() [7/7]	10
3.1.3.8 getb()	10
3.1.3.9 getd()	10
3.1.3.10 getf()	11
3.1.3.11 geti()	11
3.1.3.12 getl()	12
3.1.3.13 gets()	12
3.1.3.14 getstr()	12
<b>Index</b>	<b>15</b>



# Chapter 1

## **cxcparamx**

### 1.1 Introduction

`cxcparamx` provides C++ Extensions to `cxcparam`, CIAO's IRAF compatible parameter library

- `cxcparamx::simple` provides a simple wrapper around getting parameter values.



## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">cxcparamx::simple</a>	5
-----------------------------------	---





## Chapter 3

# Class Documentation

### 3.1 cxcparamx::simple Class Reference

```
#include <cxcparamx/simple.h>
```

#### Public Types

- using **pParamFileImpl** = std::unique\_ptr< ParamFileImpl, void(\*) (ParamFileImpl \*)>

#### Public Member Functions

- **simple** (int argc, char \*\*argv, const string &mode="rw")  
*Constructor with file name taken from argv[0].*
- **simple** (const string &filename, int argc, char \*\*argv, const string &mode="rw")  
*Constructor with explicit parameter file name.*
- void **get** (const string &name, float &value) const  
*Retrieve a parameter value into a float.*
- void **get** (const string &name, double &value) const  
*Retrieve a parameter value into a double.*
- void **get** (const string &name, short &value) const  
*Retrieve a parameter value into a short.*
- void **get** (const string &name, int &value) const  
*Retrieve a parameter value into an int.*
- void **get** (const string &name, long &value) const  
*Retrieve a parameter value into a long.*
- void **get** (const string &name, bool &value) const  
*Retrieve a parameter value into a bool.*
- void **get** (const string &name, string &value) const  
*Retrieve a parameter value into a string.*
- float **getf** (const string &name) const

*Return a parameter value as a float.*

- double `getd` (const string &name) const

*Return a parameter value as a double.*

- short `gets` (const string &name) const

*Return a parameter value as a short.*

- int `geti` (const string &name) const

*Return a parameter value as an int.*

- long `getl` (const string &name) const

*Return a parameter value as a long.*

- bool `getb` (const string &name) const

*Return a parameter value as a bool.*

- string `getstr` (const string &name) const

*Return a parameter value as a string.*

- bool `access` (const string &name) const

### 3.1.1 Detailed Description

A simple wrapper class providing polymorphic attribute retrieval.

Instantiating the class opens the parameter file; the file is closed during object destruction.

### 3.1.2 Constructor & Destructor Documentation

#### 3.1.2.1 `simple()` [1/2]

```
cxtparamx::simple::simple (
    int argc,
    char ** argv,
    const string & mode = "rw" )
```

Constructor with file name taken from argv[0].

#### Parameters

in	<i>argc</i>	the number of elements in argv
in	<i>argv</i>	typically, the command line arguments vector passed to main*
in	<i>mode</i>	mode with which to open the file

## 3.1.2.2 simple() [2/2]

```
cxtparamx::simple::simple (
    const string & in_filename,
    int argc,
    char ** argv,
    const string & mode = "rw" )
```

Constructor with explicit parameter file name.

## Parameters

in	<i>filename</i>	the parameter file name
in	<i>argc</i>	the number of elements in <i>argv</i>
	<i>argv</i>	typically, the command line arguments vector passed to <code>main</code>
in	<i>mode</i>	mode with which to open the file

## 3.1.3 Member Function Documentation

## 3.1.3.1 get() [1/7]

```
void cxtparamx::simple::get (
    const string & name,
    float & value ) const
```

Retrieve a parameter value into a float.

## Parameters

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

## Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

## 3.1.3.2 get() [2/7]

```
void cxtparamx::simple::get (
    const string & name,
    double & value ) const
```

Retrieve a parameter value into a double.

#### Parameters

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

#### 3.1.3.3 `get()` [3/7]

```
void cxcparamx::simple::get (  
    const string & name,  
    short & value ) const
```

Retrieve a parameter value into a short.

#### Parameters

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

#### 3.1.3.4 `get()` [4/7]

```
void cxcparamx::simple::get (  
    const string & name,  
    int & value ) const
```

Retrieve a parameter value into an int.

#### Parameters

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

**Exceptions**

<i>Exception</i>	if parameter is not found
------------------	---------------------------

**3.1.3.5 get()** [5/7]

```
void cxcparamx::simple::get (
    const string & name,
    long & value ) const
```

Retrieve a parameter value into a long.

**Parameters**

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

**Exceptions**

<i>Exception</i>	if parameter is not found
------------------	---------------------------

**3.1.3.6 get()** [6/7]

```
void cxcparamx::simple::get (
    const string & name,
    bool & value ) const
```

Retrieve a parameter value into an bool.

**Parameters**

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

**Exceptions**

<i>Exception</i>	if parameter is not found
------------------	---------------------------

### 3.1.3.7 `get()` [7/7]

```
void cxcparamx::simple::get (
    const string & name,
    string & value ) const
```

Retrieve a parameter value into an string.

#### Parameters

in	<i>name</i>	parameter name
out	<i>value</i>	returned value

#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

### 3.1.3.8 `getb()`

```
bool cxcparamx::simple::getb (
    const string & name ) const
```

Return a parameter value as an bool.

#### Parameters

in	<i>name</i>	parameter name
----	-------------	----------------

#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

### 3.1.3.9 `getd()`

```
double cxcparamx::simple::getd (
    const string & name ) const
```

Return a parameter value as a double.

**Parameters**

in	<i>name</i>	parameter name
----	-------------	----------------

**Exceptions**

<i>Exception</i>	if parameter is not found
------------------	---------------------------

**3.1.3.10 getf()**

```
float cxcparamx::simple::getf (
    const string & name ) const
```

Return a parameter value as a float.

**Parameters**

in	<i>name</i>	parameter name
----	-------------	----------------

**Exceptions**

<i>Exception</i>	if parameter is not found
------------------	---------------------------

**3.1.3.11 geti()**

```
int cxcparamx::simple::geti (
    const string & name ) const
```

Return a parameter value as an int.

**Parameters**

in	<i>name</i>	parameter name
----	-------------	----------------

**Exceptions**

<i>Exception</i>	if parameter is not found
------------------	---------------------------

### 3.1.3.12 getl()

```
long cxcparamx::simple::getl (
    const string & name ) const
```

Return a parameter value as a long.

#### Parameters

in	<i>name</i>	parameter name
----	-------------	----------------

#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

### 3.1.3.13 gets()

```
short cxcparamx::simple::gets (
    const string & name ) const
```

Return a parameter value as a short.

#### Parameters

in	<i>name</i>	parameter name
----	-------------	----------------

#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

### 3.1.3.14 getstr()

```
string cxcparamx::simple::getstr (
    const string & name ) const
```

Return a parameter value as an string.

#### Parameters

in	<i>name</i>	parameter name
----	-------------	----------------



#### Exceptions

<i>Exception</i>	if parameter is not found
------------------	---------------------------

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

- simple.h
- simple.cc



# Index

cxcpamx::simple, [5](#)

  get, [7–9](#)

  getb, [10](#)

  getd, [10](#)

  getf, [11](#)

  geti, [11](#)

  getl, [11](#)

  gets, [12](#)

  getstr, [12](#)

  simple, [6](#)

get

  cxcpamx::simple, [7–9](#)

getb

  cxcpamx::simple, [10](#)

getd

  cxcpamx::simple, [10](#)

getf

  cxcpamx::simple, [11](#)

geti

  cxcpamx::simple, [11](#)

getl

  cxcpamx::simple, [11](#)

gets

  cxcpamx::simple, [12](#)

getstr

  cxcpamx::simple, [12](#)

simple

  cxcpamx::simple, [6](#)