

My Project

Generated by Doxygen 1.8.11

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	Archer Class Reference	5
3.1.1	Member Function Documentation	6
3.1.1.1	Damage(int &move, int cx, int cy) override	6
3.1.1.2	Move(int &mp) override	6
3.2	Church Class Reference	6
3.2.1	Detailed Description	8
3.3	ChurchA Class Reference	8
3.3.1	Detailed Description	9
3.4	ChurchD Class Reference	9
3.4.1	Detailed Description	11
3.5	CUnit Class Reference	11
3.5.1	Member Function Documentation	13
3.5.1.1	churchEffect(int tx, int ty)	13
3.5.1.2	Damage(int &move, int x, int y)=0	13
3.5.1.3	Death()	13
3.5.1.4	findChurche(int x, int y)	13
3.5.1.5	getAWL() const	13

3.5.1.6	getDWL() const	13
3.5.1.7	getHealth() const	14
3.5.1.8	getInfo() const	14
3.5.1.9	getMaxHealth() const	14
3.5.1.10	getName() const	14
3.5.1.11	getSymbol() const	14
3.5.1.12	getX() const	14
3.5.1.13	getY() const	15
3.5.1.14	healing()	15
3.5.1.15	look()	15
3.5.1.16	Move(int &mp)=0	15
3.5.1.17	setArmy(vector< CUnit * > *a)	15
3.5.1.18	setBMask(vector< vector< IBuilding * >> *bm)	15
3.5.1.19	setMask(vector< vector< CUnit * >> *m)	16
3.5.1.20	setPlace(int x, int y)	16
3.5.1.21	TakeDamage(int damage)	16
3.5.1.22	takeOrders()	16
3.5.1.23	unlook()	16
3.6	Dwarves Class Reference	17
3.7	Elves Class Reference	18
3.8	Factory Class Reference	19
3.8.1	Detailed Description	19
3.9	GameMap Class Reference	19
3.9.1	Detailed Description	19
3.10	Humans Class Reference	20
3.11	IBuilding Class Reference	21
3.11.1	Detailed Description	22
3.11.2	Member Function Documentation	22
3.11.2.1	getPower()	22
3.11.2.2	getPurpose()	22

3.12	IRace Class Reference	22
3.12.1	Detailed Description	23
3.13	Map Class Reference	24
3.13.1	Detailed Description	24
3.13.2	Constructor & Destructor Documentation	25
3.13.2.1	Map()	25
3.13.3	Member Function Documentation	25
3.13.3.1	getSize() const	25
3.13.3.2	showMap() const	25
3.14	Spy Class Reference	25
3.14.1	Member Function Documentation	26
3.14.1.1	Damage(int &move, int cx, int cy) override	26
3.14.1.2	Move(int &mp) override	26
3.15	Warrior Class Reference	27
3.15.1	Member Function Documentation	28
3.15.1.1	Damage(int &move, int cx, int cy) override	28
3.15.1.2	Move(int &mp) override	28
3.16	World Class Reference	28
3.16.1	Detailed Description	28
3.16.2	Constructor & Destructor Documentation	29
3.16.2.1	World()	29
3.16.3	Member Function Documentation	29
3.16.3.1	getMyUnit(int n) const	29
3.16.3.2	healing() const	29
3.16.3.3	war()	29
3.17	Writer Class Reference	29
3.17.1	Detailed Description	30
3.17.2	Member Function Documentation	30
3.17.2.1	wChoise(int movepoints)	30
3.17.2.2	wDeath(string &UnitName)	30
3.17.2.3	wHealing(string &UnitName)	30
3.17.2.4	wLine()	31
3.17.2.5	wMistake()	31
3.17.2.6	wMoveInfoSpy()	31
3.17.2.7	wMoveInfoWarriorArcher()	31
3.17.2.8	wTakeCoordinates()	31
3.17.2.9	wWrongInfo()	31

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

CUnit	11
Archer	5
Spy	25
Warrior	27
Factory	19
GameMap	19
IBuilding	21
Church	6
ChurchA	8
ChurchD	9
IRace	22
Dwarves	17
Elves	18
Humans	20
Map	24
World	28
Writer	29

Chapter 2

Class Index

2.1 Class List

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

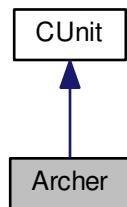
Archer	5
Church	
Buffs attack or defence	6
ChurchA	
Buffs attack	8
ChurchD	
Buffs defence	9
CUnit	11
Dwarves	17
Elves	18
Factory	
Unit's factory	19
GameMap	
Singleton class. Makes gamemap	19
Humans	20
IBuilding	
Building's interface	21
IRace	
Interface of races	22
Map	
Map text image	24
Spy	25
Warrior	27
World	
Makes the gameworld. Initializes maps with objects	28
Writer	
This class is for speaking with player	29

Chapter 3

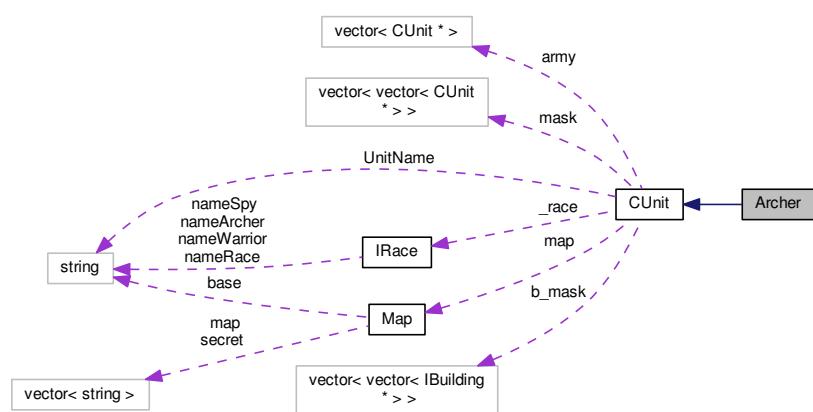
Class Documentation

3.1 Archer Class Reference

Inheritance diagram for Archer:



Collaboration diagram for Archer:



Public Member Functions

- **Archer** (*IRace *race, Map *m, bool ie*)
- void **Move** (*int &mp*) override
- void **Damage** (*int &move, int cx, int cy*) override

Additional Inherited Members

3.1.1 Member Function Documentation

3.1.1.1 void Archer::Damage (*int & move, int x, int y*) [inline], [override], [virtual]

hit the enemy

Parameters

<i>move</i>	way to enemy on (x,y)
<i>x</i>	
<i>y</i>	

Implements [CUnit](#).

3.1.1.2 void Archer::Move (*int & mp*) [inline], [override], [virtual]

your unit's moving

Parameters

<i>mp</i>	movepoints
-----------	------------

Implements [CUnit](#).

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

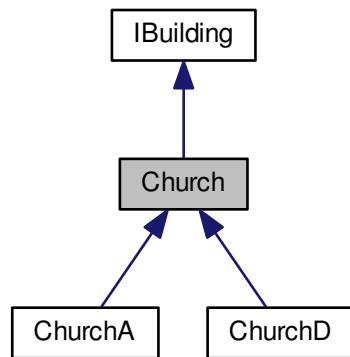
- Units.h

3.2 Church Class Reference

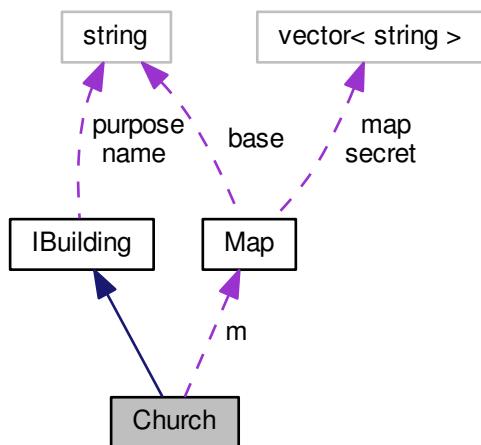
buffs attack or defence

```
#include <Buildings.h>
```

Inheritance diagram for Church:



Collaboration diagram for Church:



Public Member Functions

- `Church (int x, int y, Map *m, int p)`

Protected Attributes

- `int x`
- `int y`
- `Map * m`

3.2.1 Detailed Description

buffs attack or defence

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

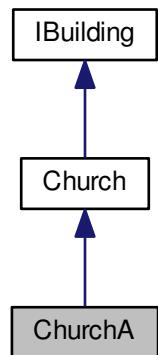
- Buildings.h

3.3 ChurchA Class Reference

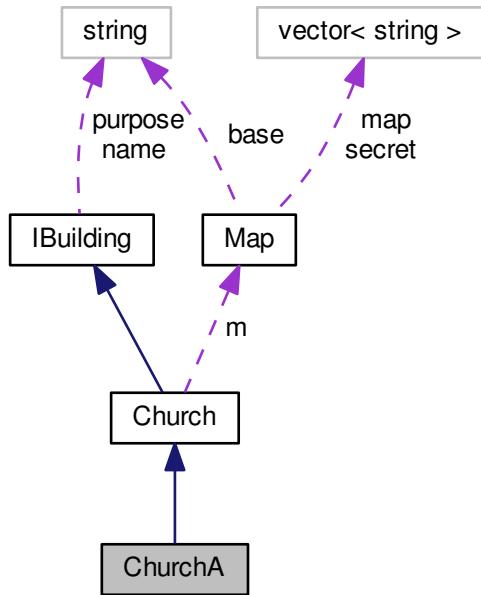
buffs attack

```
#include <Buildings.h>
```

Inheritance diagram for ChurchA:



Collaboration diagram for ChurchA:



Public Member Functions

- `ChurchA (int x, int y, Map *m, int p)`

Additional Inherited Members

3.3.1 Detailed Description

buffs attack

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

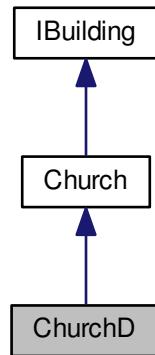
- `Buildings.h`

3.4 ChurchD Class Reference

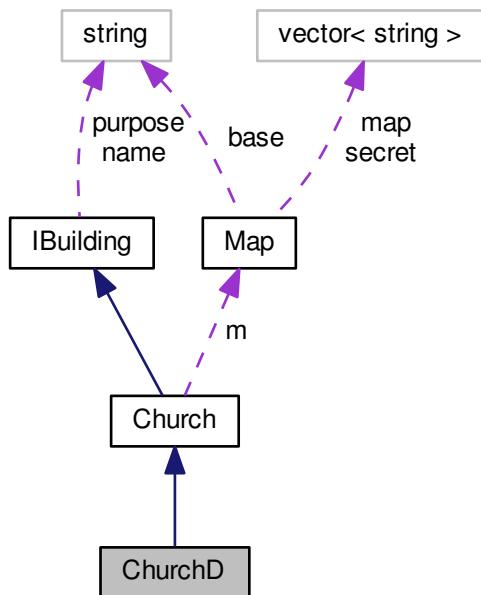
buffs defence

```
#include <Buildings.h>
```

Inheritance diagram for ChurchD:



Collaboration diagram for ChurchD:



Public Member Functions

- **ChurchD** (int x, int y, [Map](#) *m, int p)

Additional Inherited Members

3.4.1 Detailed Description

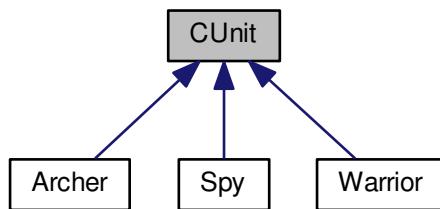
buffs defence

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

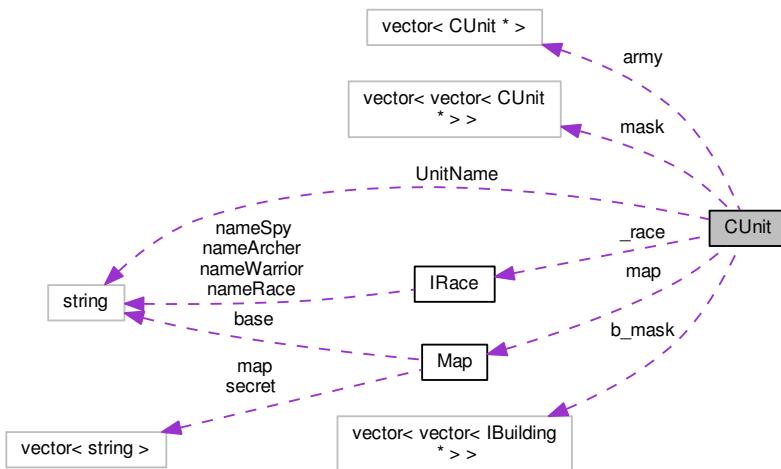
- Buildings.h

3.5 CUnit Class Reference

Inheritance diagram for CUnit:



Collaboration diagram for CUnit:



Public Member Functions

- **CUnit** (IRace *r, Map *map)
- virtual void **Move** (int &mp)=0
- virtual void **Damage** (int &move, int x, int y)=0
- void **TakeDamage** (int damage)
- void **getInfo** () const
- string **getName** () const
- void **takeOrders** ()
- void **setPlace** (int x, int y)
- char **getSymbol** () const
- int **getX** () const
- int **getY** () const
- void **healing** ()
- int **getHealth** () const
- int **getMaxHealth** () const
- void **setArmy** (vector< CUnit * > *a)
- void **setMask** (vector< vector< CUnit * >> *m)
- void **setBMask** (vector< vector< IBuilding * >> *bm)
- int **getAWL** () const
- int **getDWL** () const

Protected Member Functions

- void **Death** ()
- void **churchEffect** (int tx, int ty)
- void **findChurche** (int x, int y)
- void **look** ()
- void **unlock** ()

Protected Attributes

- **IRace * _race**
- vector< CUnit * > * **army**
- vector< vector< CUnit * >> * **mask**
- vector< vector< IBuilding * >> * **b_mask**
- string **UnitName**
- int **move**
- int **defence**
- int **attack**
- int **attackDistance** = 1
- int **lookDistance**
- int **health**
- int **maxhealth**
- int **attackWeaponLevel** = 0
- int **defenceWeaponLevel** = 0
- int **x**
- int **y**
- char **symbol**
- Map * **map**
- bool **isEnemy**
- int **id**
- int **powerWeaponLevel** = 5

3.5.1 Member Function Documentation

3.5.1.1 void CUnit::churchEffect(int *tx*, int *ty*) [inline], [protected]

check (tx,ty) church's effect's

Parameters

<i>tx</i>	
<i>ty</i>	

3.5.1.2 virtual void CUnit::Damage(int & *move*, int *x*, int *y*) [pure virtual]

hit the enemy

Parameters

<i>move</i>	way to enemy on (x,y)
<i>x</i>	
<i>y</i>	

Implemented in [Spy](#), [Archer](#), and [Warrior](#).

3.5.1.3 void CUnit::Death() [inline], [protected]

Unit's death: deleting from maps

3.5.1.4 void CUnit::findChurche(int *x*, int *y*) [inline], [protected]

check is there a churche

Parameters

<i>x</i>	
<i>y</i>	

3.5.1.5 int CUnit::getAWL() const [inline]

Returns

weapon's attack level

3.5.1.6 int CUnit::getDWL() const [inline]

Returns

weapon's defence level

3.5.1.7 int CUnit::getHealth () const [inline]**Returns**

health of unit

3.5.1.8 void CUnit:: getInfo () const [inline]

say about the current unit

3.5.1.9 int CUnit::getMaxHealth () const [inline]**Returns**

maxhealth of unit

3.5.1.10 string CUnit::getName () const [inline]**Returns**

unit's name

3.5.1.11 char CUnit::getSymbol () const [inline]

get unit's mark on text map

Returns

symbol

3.5.1.12 int CUnit::getX () const [inline]

get x-coordinate

Returns

x

3.5.1.13 int CUnit::getY() const [inline]

get y-coordinate

Returns

y

3.5.1.14 void CUnit::healing() [inline]

heal unit if it possible

3.5.1.15 void CUnit::look() [inline], [protected]

change '*' to '.' in unit's area of visibility

3.5.1.16 virtual void CUnit::Move(int & mp) [pure virtual]

your unit's moving

Parameters

mp	movepoints
----	------------

Implemented in [Spy](#), [Archer](#), and [Warrior](#).

3.5.1.17 void CUnit::setArmy(vector< CUnit * > * a) [inline]

set unit's army

Parameters

a	
---	--

3.5.1.18 void CUnit::setBMask(vector< vector< IBuilding * >> * bm) [inline]

set map with buildings

Parameters

bm	
----	--

3.5.1.19 void CUnit::setMask (`vector<vector< CUnit * >> * m`) [inline]

set map with units

Parameters

<code>m</code>	
----------------	--

3.5.1.20 void CUnit::setPlace (`int x, int y`) [inline]

set place for unit

Parameters

<code>x</code>	
<code>y</code>	

3.5.1.21 void CUnit::TakeDamage (`int damage`) [inline]

count the damage to unit's health if it bigger than health, unit will die

Parameters

<code>damage</code>	damage that your unit get from hit
---------------------	------------------------------------

3.5.1.22 void CUnit::takeOrders () [inline]

unit recieve orders from you

- move
- fight
- nothing

3.5.1.23 void CUnit::unlock() [inline], [protected]

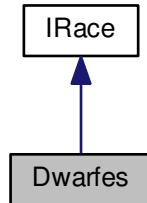
change '.' to '*' in unit's area of visibility when he moves

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

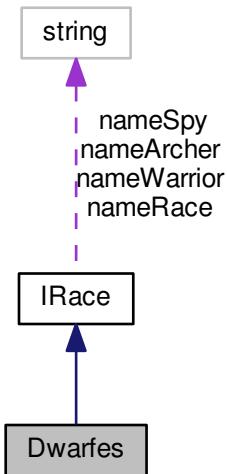
- Units.h

3.6 Dwarves Class Reference

Inheritance diagram for Dwarves:



Collaboration diagram for Dwarves:



Public Member Functions

- void **superAction ()** override

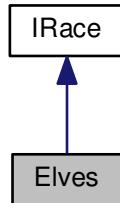
Additional Inherited Members

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

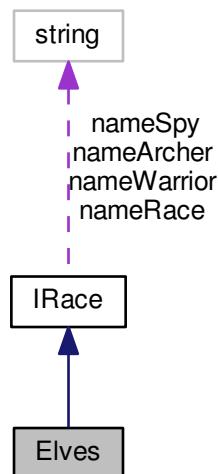
- Race.h

3.7 Elves Class Reference

Inheritance diagram for Elves:



Collaboration diagram for Elves:



Public Member Functions

- void **superAction ()** override

Additional Inherited Members

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

- Race.h

3.8 Factory Class Reference

Unit's factory.

```
#include <Units.h>
```

Static Public Member Functions

- static **CUnit** * **create** (string type, **IRace** *race, **Map** *m, bool ie)

3.8.1 Detailed Description

Unit's factory.

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

- Units.h

3.9 GameMap Class Reference

Singleton class. Makes gamemap.

```
#include <World.h>
```

Public Member Functions

- **GameMap** (const **GameMap** &)=delete
- **GameMap** & **operator=** (const **GameMap** &)=delete

Static Public Member Functions

- static **Map** * **instance** ()

3.9.1 Detailed Description

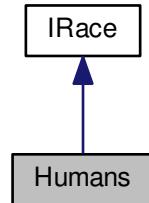
Singleton class. Makes gamemap.

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

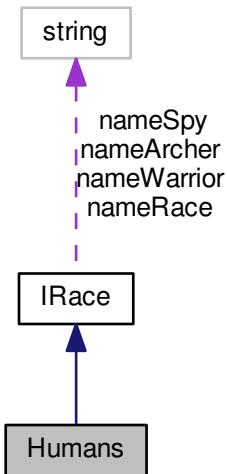
- World.h

3.10 Humans Class Reference

Inheritance diagram for Humans:



Collaboration diagram for Humans:



Public Member Functions

- void **superAction ()** override

Additional Inherited Members

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

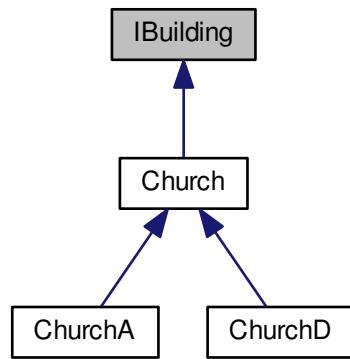
- Race.h

3.11 IBuilding Class Reference

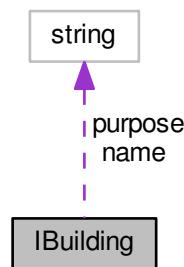
building's interface

```
#include <Buildings.h>
```

Inheritance diagram for IBuilding:



Collaboration diagram for IBuilding:



Public Member Functions

- `string getPurpose ()`
- `int getPower ()`

Protected Attributes

- int **high**
- int **wall**
- string **name**
- string **purpose**
- int **power**

3.11.1 Detailed Description

building's interface

3.11.2 Member Function Documentation

3.11.2.1 int IBuilding::getPower() [inline]

Returns

purpose(bonus) that unit can take staying near the building

3.11.2.2 string IBuilding::getPurpose() [inline]

Returns

goal of building, defence or attack

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

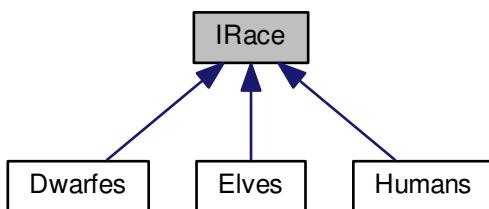
- Buildings.h

3.12 IRace Class Reference

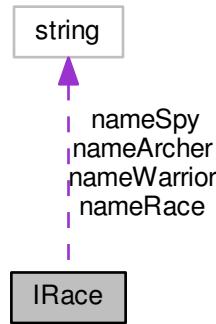
Interface of races.

```
#include <Race.h>
```

Inheritance diagram for IRace:



Collaboration diagram for IRace:



Public Member Functions

- string **getNameRace** () const
- string **getNameArcher** () const
- string **getNameWarrior** () const
- string **getNameSpy** () const
- int **getBonusMove** () const
- int **getBonusAttack** () const
- int **getBonusDefence** () const
- int **getBonusArcherDistance** () const
- int **getBonusLookDistance** () const
- int **getBonusHealth** () const
- virtual void **superAction** ()=0

Protected Attributes

- string **nameRace**
- string **nameArcher**
- string **nameWarrior**
- string **nameSpy**
- int **bonusMove**
- int **bonusAttack**
- int **bonusDefence**
- int **bonusArcherDistance**
- int **bonusLookDistance**
- int **bonusHealth**

3.12.1 Detailed Description

Interface of races.

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

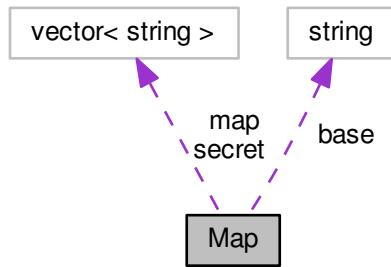
- Race.h

3.13 Map Class Reference

[Map](#) text image.

```
#include <Map.h>
```

Collaboration diagram for Map:



Public Member Functions

- [Map \(\)](#)
- void [showMap \(\) const](#)
- int [getSize \(\) const](#)

Public Attributes

- string **base**
- vector< string > **map**
- vector< string > **secret**

3.13.1 Detailed Description

[Map](#) text image.

[Map](#) is showing like number of symbols.

- * - unknown place
- . - place in your area of visibility
- **A - Z** - your units
- **a - z** - enemy units
- + - church

3.13.2 Constructor & Destructor Documentation

3.13.2.1 Map::Map() [inline]

initialize map with '*'

3.13.3 Member Function Documentation

3.13.3.1 int Map::getSize() const [inline]

Returns

map's size

3.13.3.2 void Map::showMap() const [inline]

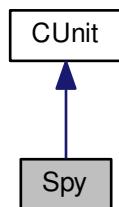
show the map

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

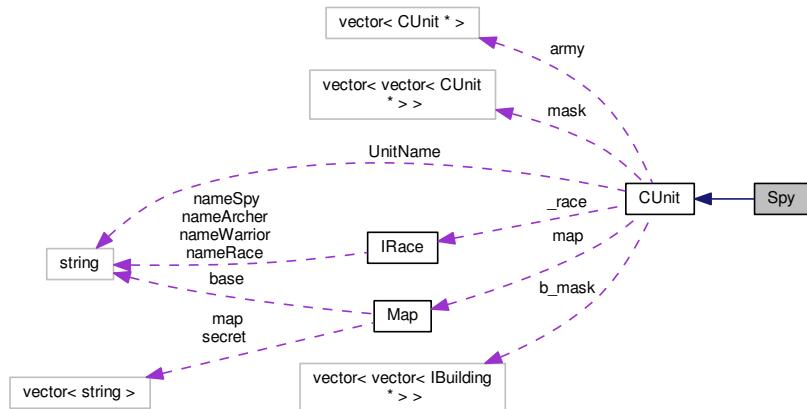
- Map.h

3.14 Spy Class Reference

Inheritance diagram for Spy:



Collaboration diagram for Spy:



Public Member Functions

- **Spy** (*IRace* *race, *Map* *m, bool ie)
- void **Move** (int &*mp*) override
- void **Damage** (int &*move*, int cx, int cy) override

Additional Inherited Members

3.14.1 Member Function Documentation

3.14.1.1 void Spy::Damage (int & move, int x, int y) [inline], [override], [virtual]

hit the enemy

Parameters

<i>move</i>	way to enemy on (x,y)
<i>x</i>	
<i>y</i>	

Implements [CUnit](#).

3.14.1.2 void Spy::Move (int & mp) [inline], [override], [virtual]

your unit's moving

Parameters

<i>mp</i>	movepoints
-----------	------------

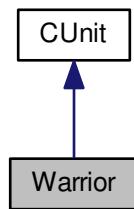
Implements [CUnit](#).

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

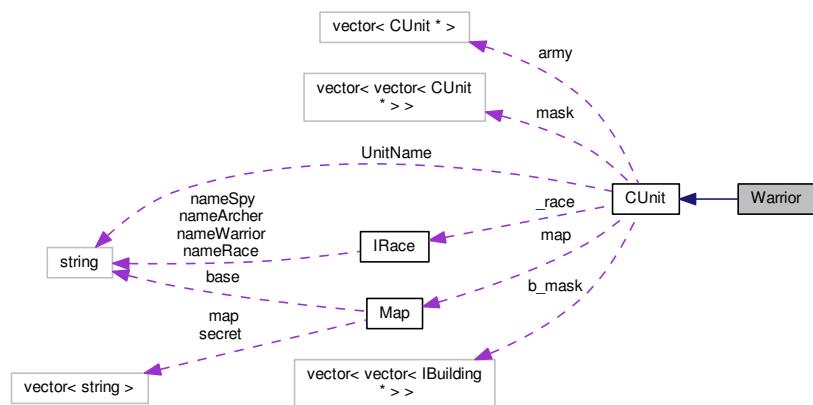
- Units.h

3.15 Warrior Class Reference

Inheritance diagram for Warrior:



Collaboration diagram for Warrior:



Public Member Functions

- **Warrior** ([IRace](#) *race, [Map](#) *m, bool ie)
- void [Move](#) (int &mp) override
- void [Damage](#) (int &move, int cx, int cy) override

Additional Inherited Members

3.15.1 Member Function Documentation

3.15.1.1 void Warrior::Damage (int & move, int x, int y) [inline], [override], [virtual]

hit the enemy

Parameters

<i>move</i>	way to enemy on (x,y)
<i>x</i>	
<i>y</i>	

Implements [CUnit](#).

3.15.1.2 void Warrior::Move (int & mp) [inline], [override], [virtual]

your unit's moving

Parameters

<i>mp</i>	movepoints
-----------	------------

Implements [CUnit](#).

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

- Units.h

3.16 World Class Reference

Makes the gameworld. Initializes maps with objects.

```
#include <World.h>
```

Public Member Functions

- [World \(\)](#)
- [CUnit * getMyUnit \(int n\) const](#)
- [void healing \(\) const](#)
- [bool war \(\)](#)

3.16.1 Detailed Description

Makes the gameworld. Initializes maps with objects.

3.16.2 Constructor & Destructor Documentation

3.16.2.1 `World::World() [inline]`

Initializing

3.16.3 Member Function Documentation

3.16.3.1 `CUnit* World::getMyUnit(int n) const [inline]`

Parameters

<code>n</code>	unit's number
----------------	---------------

Returns

unit

3.16.3.2 `void World::healing() const [inline]`

Heal alive units

3.16.3.3 `bool World::war() [inline]`

One step of the game. Every alive unit takes orders. If there nobody alive, game ends.

Returns

is the game ended

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

- `World.h`

3.17 Writer Class Reference

this class is for speaking with player

```
#include <Units.h>
```

Static Public Member Functions

- static void `wChoise` (int movepoints)
- static void `wTakeCoordinates` ()
- static void `wWrongInfo` ()
- static void `wDeath` (string &UnitName)
- static void `wDamage` (string &UnitName, int damage)
- static void `wHealing` (string &UnitName)
- static void `wMoveInfoWarriorArcher` ()
- static void `wMistake` ()
- static void `wMoveInfoSpy` ()
- static void `wLine` ()

3.17.1 Detailed Description

this class is for speaking with player

3.17.2 Member Function Documentation

3.17.2.1 static void Writer::wChoise (int *movepoints*) [inline], [static]

ask you to order something

Parameters

<code>movepoints</code>	current unit movepoints
-------------------------	-------------------------

3.17.2.2 static void Writer::wDeath (string & *UnitName*) [inline], [static]

say that this unit has dead

Parameters

<code>UnitName</code>	unit's name
-----------------------	-------------

3.17.2.3 static void Writer::wHealing (string & *UnitName*) [inline], [static]

say that your unit is healing

Parameters

<code>UnitName</code>	
-----------------------	--

3.17.2.4 **static void Writer::wLine() [inline], [static]**

print 2 blank lines

3.17.2.5 **static void Writer::wMistake() [inline], [static]**

say about an error

3.17.2.6 **static void Writer::wMoveInfoSpy() [inline], [static]**

ask you to order the way

3.17.2.7 **static void Writer::wMoveInfoWarriorArcher() [inline], [static]**

ask you to order the way

3.17.2.8 **static void Writer::wTakeCoordinates() [inline], [static]**

ask you coordinates

3.17.2.9 **static void Writer::wWrongInfo() [inline], [static]**

say about error

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

- Units.h

Index

Archer, 5
 Damage, 6
 Move, 6

CUnit, 11
 churchEffect, 13
 Damage, 13
 Death, 13
 findChurche, 13
 getAWL, 13
 getDWL, 13
 getHealth, 14
 getInfo, 14
 getMaxHealth, 14
 getName, 14
 getSymbol, 14
 getX, 14
 getY, 14
 healing, 15
 look, 15
 Move, 15
 setArmy, 15
 setBMask, 15
 setMask, 15
 setPlace, 16
 TakeDamage, 16
 takeOrders, 16
 unlook, 16

Church, 6
churchEffect
 CUnit, 13

ChurchA, 8
ChurchD, 9

Damage
 Archer, 6
 CUnit, 13
 Spy, 26
 Warrior, 28

Death
 CUnit, 13

Dwarfes, 17

Elves, 18

Factory, 19

findChurche
 CUnit, 13

GameMap, 19

getAWL

 CUnit, 13

getDWL
 CUnit, 13

getHealth
 CUnit, 14

getInfo
 CUnit, 14

getMaxHealth
 CUnit, 14

getMyUnit
 World, 29

getName
 CUnit, 14

getPower
 IBuilding, 22

getPurpose
 IBuilding, 22

getSize
 Map, 25

getSymbol
 CUnit, 14

getX
 CUnit, 14

getY
 CUnit, 14

healing
 CUnit, 15
 World, 29

Humans, 20

 IRace, 22

 IBuilding, 21
 getPower, 22
 getPurpose, 22

 look
 CUnit, 15

 Map, 24
 getSize, 25
 Map, 25
 showMap, 25

 Move
 Archer, 6
 CUnit, 15
 Spy, 26
 Warrior, 28

 setArmy
 CUnit, 15

setBMask
 CUnit, 15
setMask
 CUnit, 15
setPlace
 CUnit, 16
showMap
 Map, 25
Spy, 25
 Damage, 26
 Move, 26

TakeDamage
 CUnit, 16
takeOrders
 CUnit, 16

unlook
 CUnit, 16

wChoice
 Writer, 30
wDeath
 Writer, 30
wHealing
 Writer, 30
wLine
 Writer, 30
wMistake
 Writer, 31
wMoveInfoSpy
 Writer, 31
wMoveInfoWarriorArcher
 Writer, 31
wTakeCoordinates
 Writer, 31
wWrongInfo
 Writer, 31
war
 World, 29
Warrior, 27
 Damage, 28
 Move, 28
World, 28
 getMyUnit, 29
 healing, 29
 war, 29
 World, 29
Writer, 29
 wChoice, 30
 wDeath, 30
 wHealing, 30
 wLine, 30
 wMistake, 31
 wMoveInfoSpy, 31
 wMoveInfoWarriorArcher, 31
 wTakeCoordinates, 31
 wWrongInfo, 31