

Aqua

Generated by Doxygen 1.9.1

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 Class Documentation	5
3.1 aq::AlignmentForce Class Reference	5
3.2 aq::Breeder Class Reference	7
3.3 aq::CohesionForce Class Reference	8
3.4 aq::Engine Class Reference	10
3.5 aq::Fish Class Reference	12
3.6 aq::Force Class Reference	13
3.7 aq::Island Class Reference	15
3.8 aq::Net::LocalisedIterator Class Reference	16
3.9 aq::MinSpeedForce Class Reference	18
3.10 aq::Net Class Reference	20
3.11 shader::PerlinNoise Class Reference	22
3.11.1 Detailed Description	23
3.11.2 Member Function Documentation	24
3.11.2.1 colorFromHeight()	24
3.11.2.2 fractalNoise()	24
3.11.2.3 perlin()	24
3.11.2.4 randomGradient()	25
3.12 aq::SeparationForce Class Reference	25
3.13 aq::Breeder::Settings Struct Reference	27
3.14 aq::WaterResistantForce Class Reference	28
Index	31

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

aq::Breeder	7
aq::Engine	10
aq::Fish	12
aq::Force	13
aq::AlignmentForce	5
aq::CohesionForce	8
aq::MinSpeedForce	18
aq::SeparationForce	25
aq::WaterResistantForce	28
aq::Island	15
aq::Net::LocalisedIterator	16
aq::Net	20
shader::PerlinNoise	22
aq::Breeder::Settings	27

Chapter 2

Class Index

2.1 Class List

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

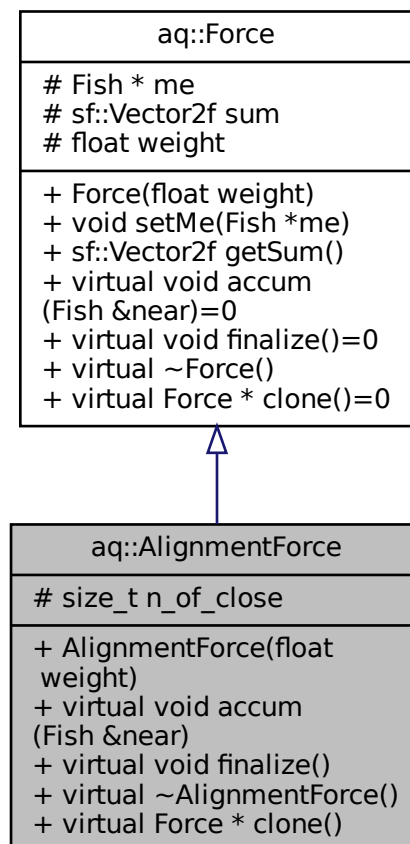
aq::AlignmentForce	5
aq::Breeder	7
aq::CohesionForce	8
aq::Engine	10
aq::Fish	12
aq::Force	13
aq::Island	15
aq::Net::LocalisedIterator	16
aq::MinSpeedForce	18
aq::Net	20
shader::PerlinNoise	
Simple 2D perlin noise shader	22
aq::SeparationForce	25
aq::Breeder::Settings	27
aq::WaterResistantForce	28

Chapter 3

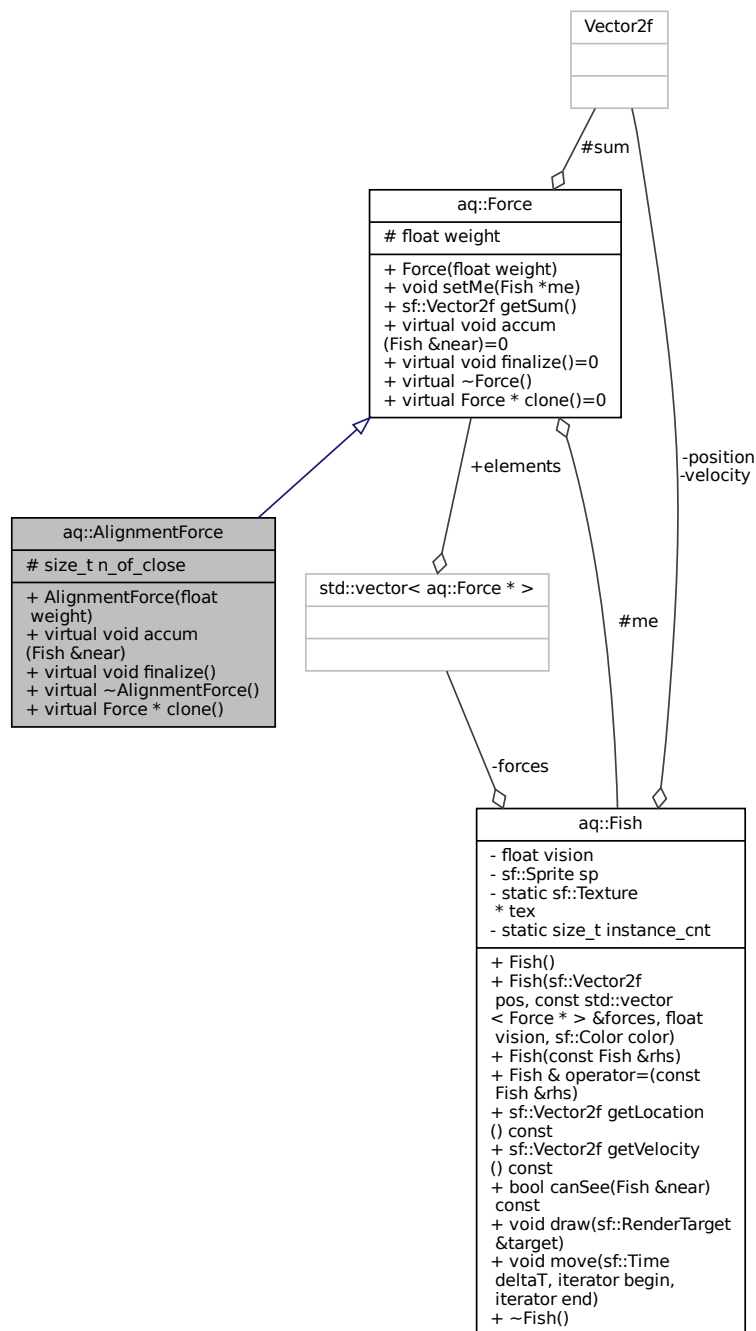
Class Documentation

3.1 aq::AlignmentForce Class Reference

Inheritance diagram for aq::AlignmentForce:



Collaboration diagram for `aq::AlignmentForce`:



Public Member Functions

- **AlignmentForce** (float weight)
- virtual void **accum** ([Fish](#) &near)
- virtual void **finalize** ()
- virtual [Force](#) * **clone** ()

Protected Attributes

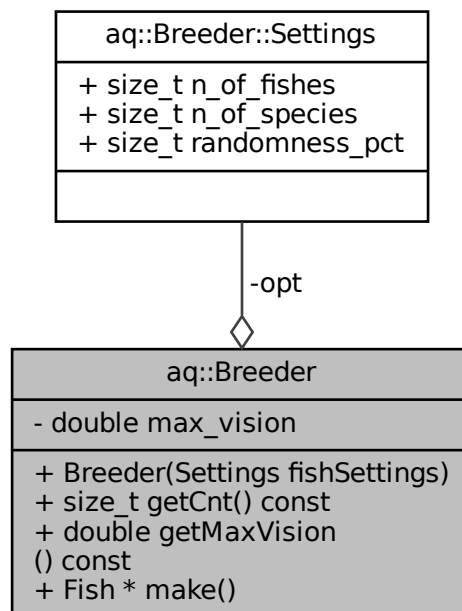
- `size_t n_of_close {0}`

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

- `inc/forces.hpp`

3.2 aq::Breeder Class Reference

Collaboration diagram for aq::Breeder:



Classes

- struct [Settings](#)

Public Member Functions

- **Breeder** ([Settings](#) fishSettings)
- `size_t getCnt () const`
- `double getMaxVision () const`
- [Fish](#) * **make** ()

Private Attributes

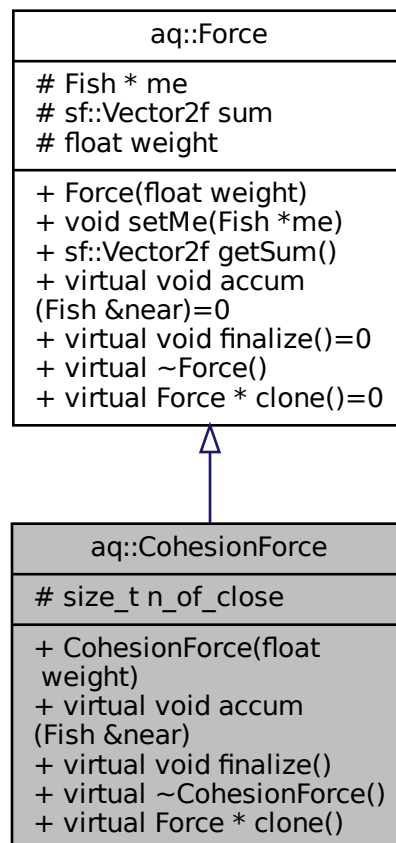
- const [Settings](#) **opt**
- double **max_vision** = 0

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

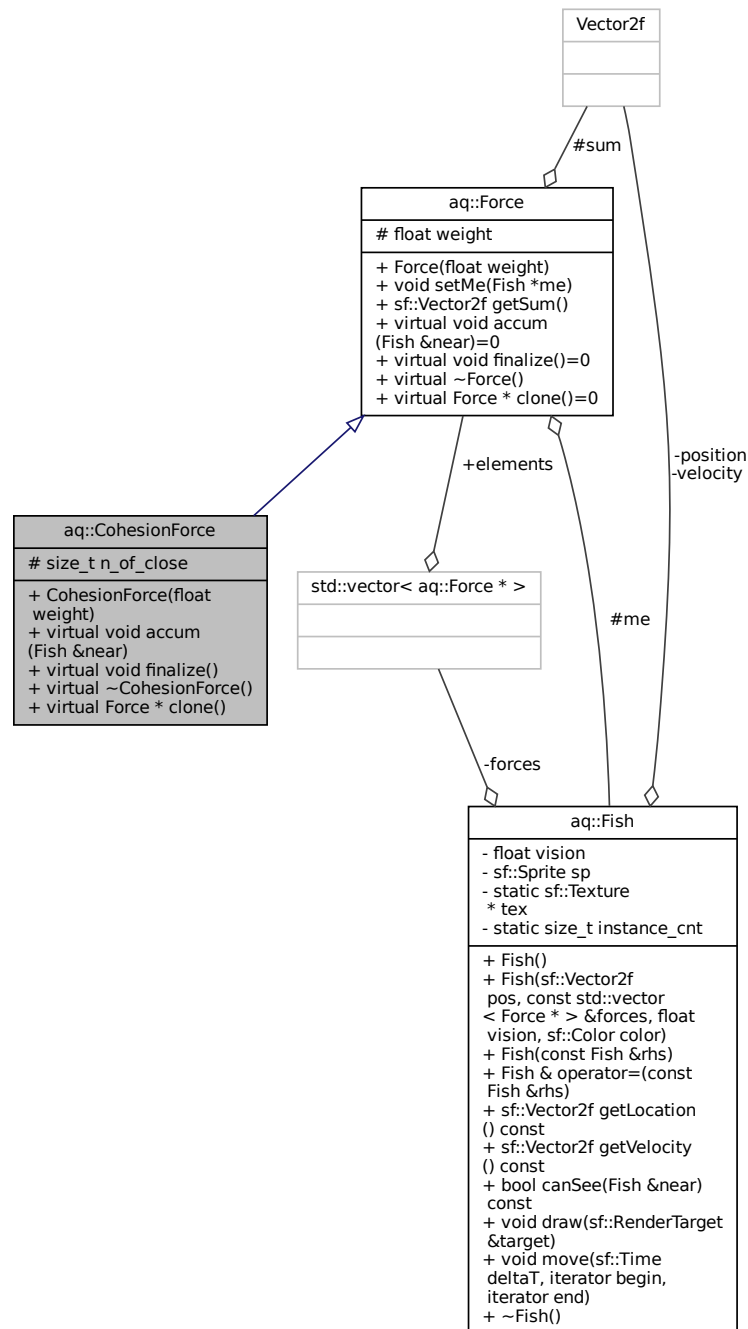
- inc/breeder.hpp
- src/breeder.cpp

3.3 aq::CohesionForce Class Reference

Inheritance diagram for aq::CohesionForce:



Collaboration diagram for aq::CohesionForce:



Public Member Functions

- **CohesionForce** (float weight)
- virtual void **accum** (Fish &near)
- virtual void **finalize** ()
- virtual Force * **clone** ()

Protected Attributes

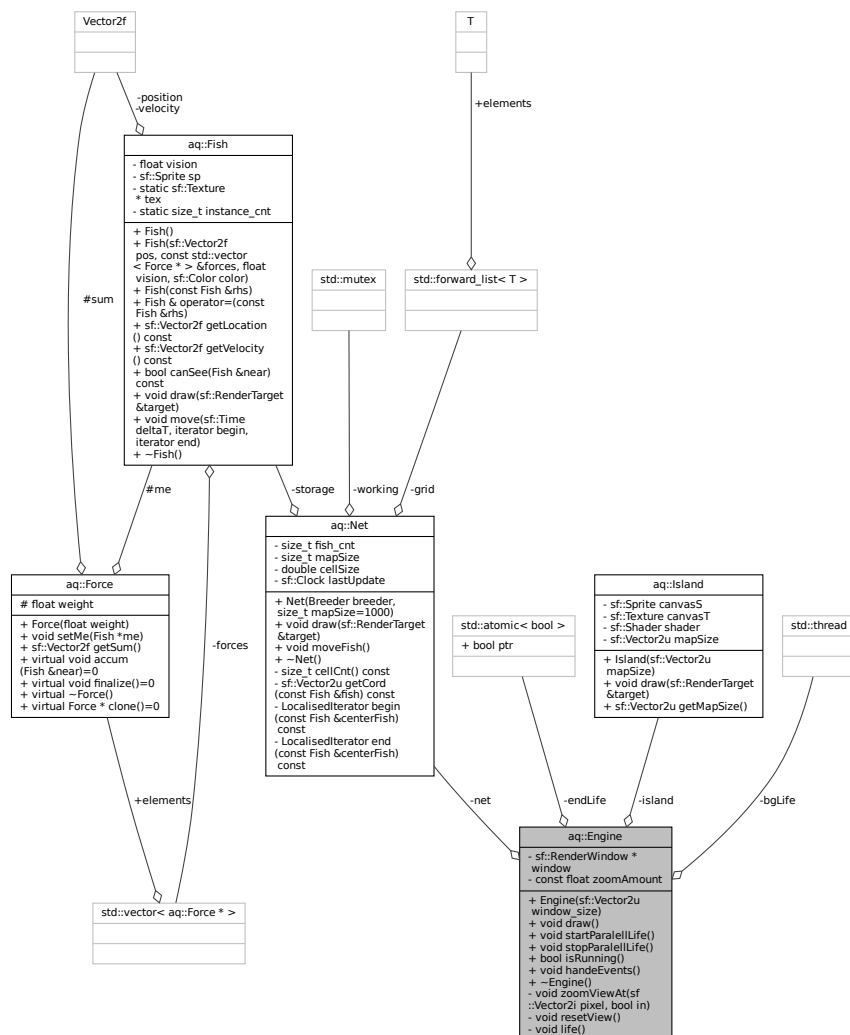
- `size_t n_of_close {0}`

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

- `inc/forces.hpp`

3.4 aq::Engine Class Reference

Collaboration diagram for `aq::Engine`:



Public Member Functions

- **Engine** (`sf::Vector2u window_size`)
- **void draw** ()
- **void startParallelLife** ()
- **void stopParallelLife** ()
- **bool isRunning** ()
- **void handleEvents** ()

Private Member Functions

- void **zoomViewAt** (sf::Vector2i pixel, bool in)
- void **resetView** ()
- void **life** ()

Private Attributes

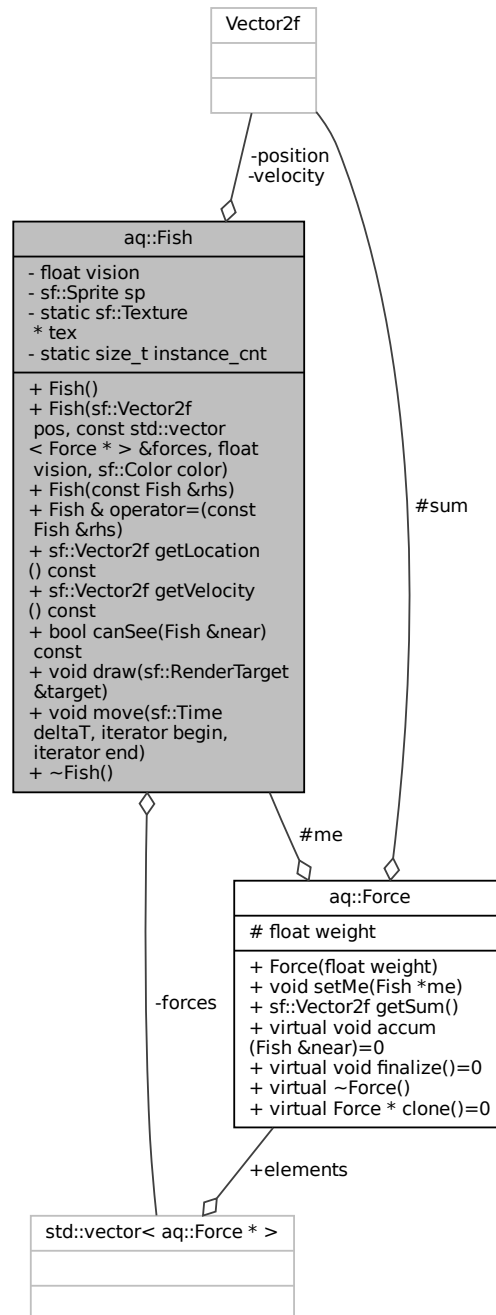
- sf::RenderWindow * **window**
- [Net](#) * **net**
- [Island](#) * **island**
- std::atomic< bool > **endLife**
- const float **zoomAmount** = 1.3F
- std::thread **bgLife**

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

- inc/engine.hpp
- src/engine.cpp

3.5 aq::Fish Class Reference

Collaboration diagram for aq::Fish:



Public Member Functions

- **Fish** (`sf::Vector2f pos`, `const std::vector< Force * > &forces`, `float vision`, `sf::Color color`)
- **Fish** (`const Fish &rhs`)

- **Fish** & **operator=** (const **Fish** &rhs)
- sf::Vector2f **getLocation** () const
- sf::Vector2f **getVelocity** () const
- bool **canSee** (**Fish** &near) const
- void **draw** (sf::RenderTarget &target)
- template<typename iterator >
void **move** (sf::Time deltaT, iterator begin, iterator end)

Private Attributes

- sf::Vector2f **position**
- sf::Vector2f **velocity**
- std::vector< **Force** * > **forces**
- float **vision**
- sf::Sprite **sp**

Static Private Attributes

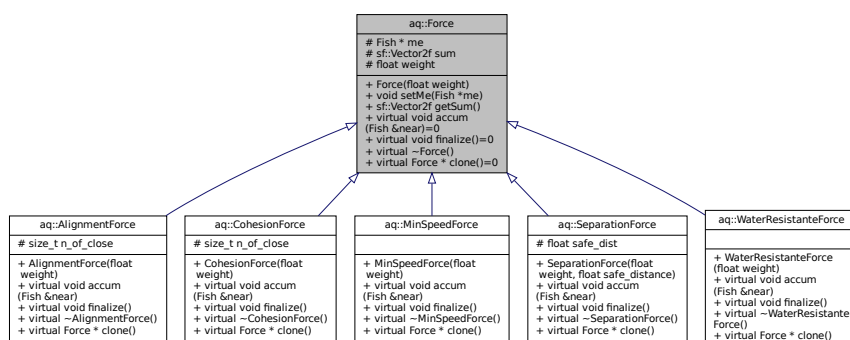
- static sf::Texture * **tex** = nullptr
- static size_t **instance_cnt** = 0

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

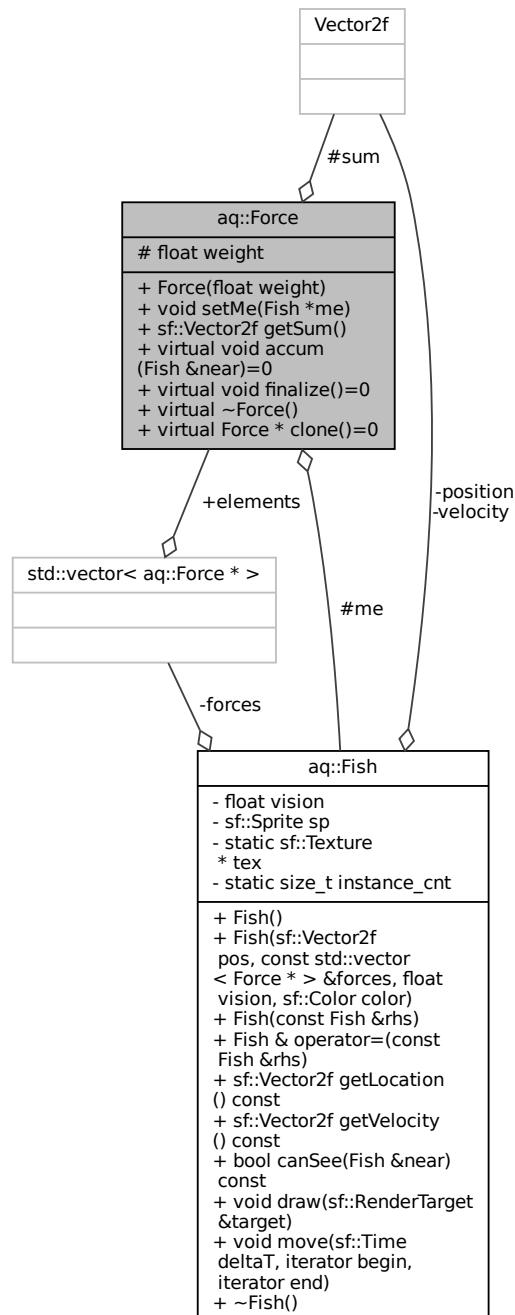
- inc/fish.hpp
- src/fish.cpp

3.6 aq::Force Class Reference

Inheritance diagram for aq::Force:



Collaboration diagram for `aq::Force`:



Public Member Functions

- **Force** (float weight)
- void **setMe** ([Fish](#) *me)
- sf::Vector2f **getSum** ()
- virtual void **accum** ([Fish](#) &near)=0
- virtual void **finalize** ()=0
- virtual [Force](#) * **clone** ()=0

Protected Attributes

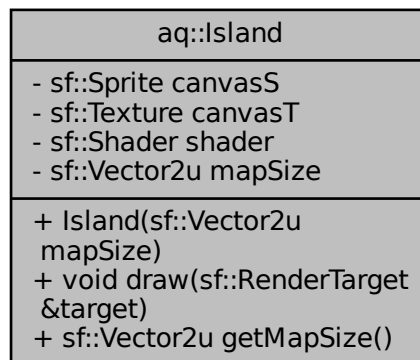
- `Fish * me {nullptr}`
- `sf::Vector2f sum {0, 0}`
- `float weight`

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

- `inc/force.hpp`
- `src/force.cpp`

3.7 aq::Island Class Reference

Collaboration diagram for aq::Island:



Public Member Functions

- `Island (sf::Vector2u mapSize)`
- `void draw (sf::RenderTarget &target)`
- `sf::Vector2u getMapSize ()`

Private Attributes

- `sf::Sprite canvasS`
- `sf::Texture canvasT`
- `sf::Shader shader`
- `sf::Vector2u mapSize`

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

- `inc/island.hpp`
- `src/island.cpp`

- [Fish](#) & **operator*** ()
- [Fish](#) * **operator->** ()
- [LocalisedIterator](#) & **operator++** ()
- [LocalisedIterator](#) **operator++** (int)
- bool **operator==** (const [LocalisedIterator](#) &rhs)

Private Member Functions

- sf::Vector2u **currCord** () const
- void **updateIters** ()
- bool **atEnd** () const

Private Attributes

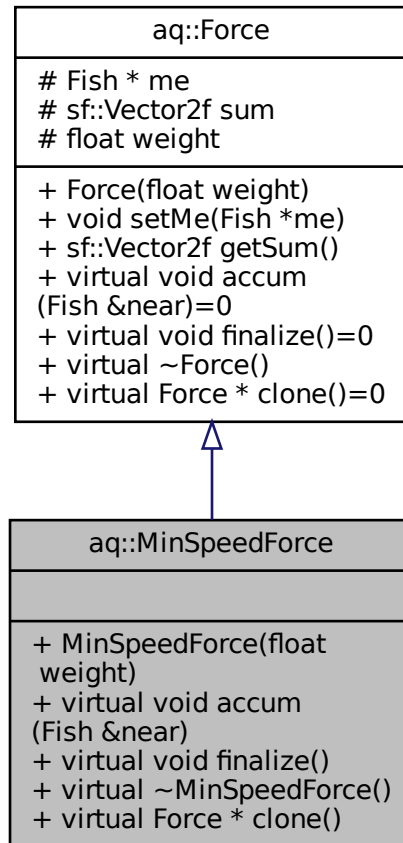
- const [Net](#) & **net**
- const sf::Vector2u **centerCord**
- cell::iterator **currIter**
- cell::iterator **currEnd**
- size_t **idx** {0}

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

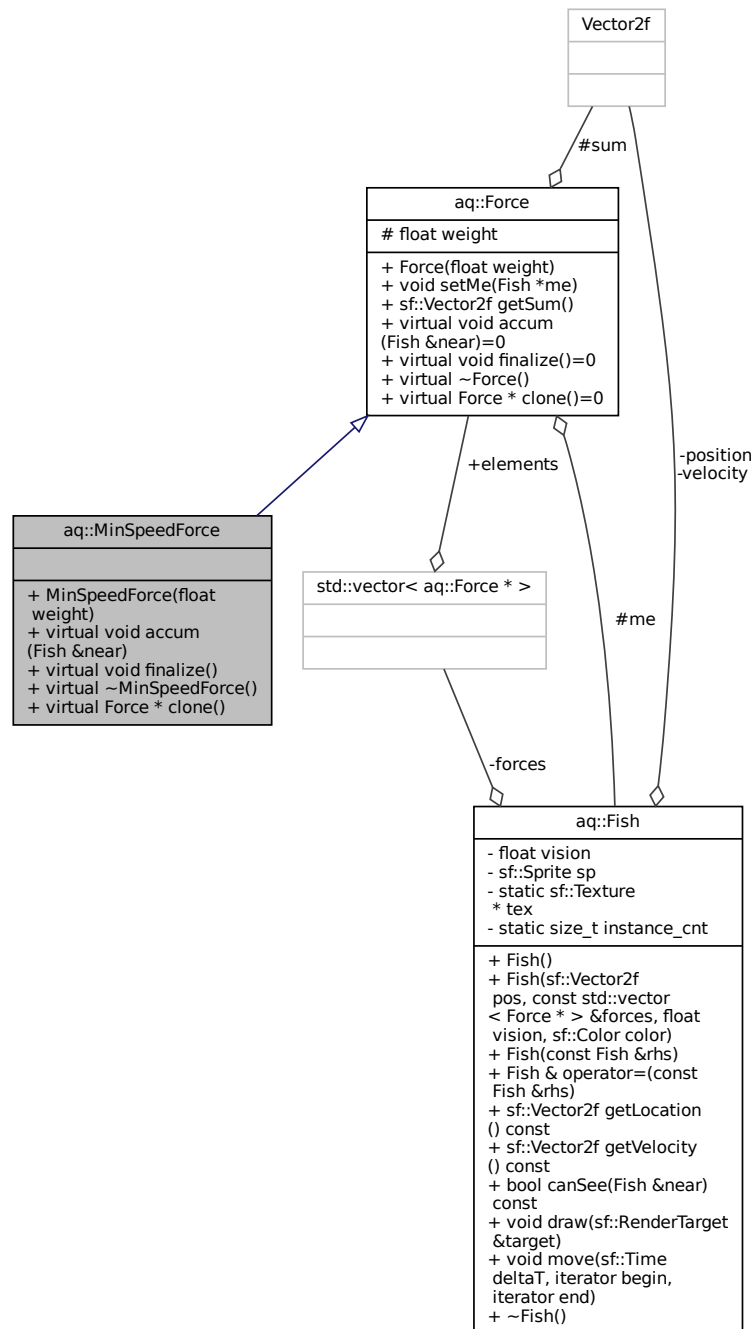
- inc/net.hpp
- src/iter.cpp

3.9 aq::MinSpeedForce Class Reference

Inheritance diagram for aq::MinSpeedForce:



Collaboration diagram for aq::MinSpeedForce:



Public Member Functions

- **MinSpeedForce** (float weight)
- virtual void **accum** ([Fish](#) &near)
- virtual void **finalize** ()
- virtual [Force](#) * **clone** ()

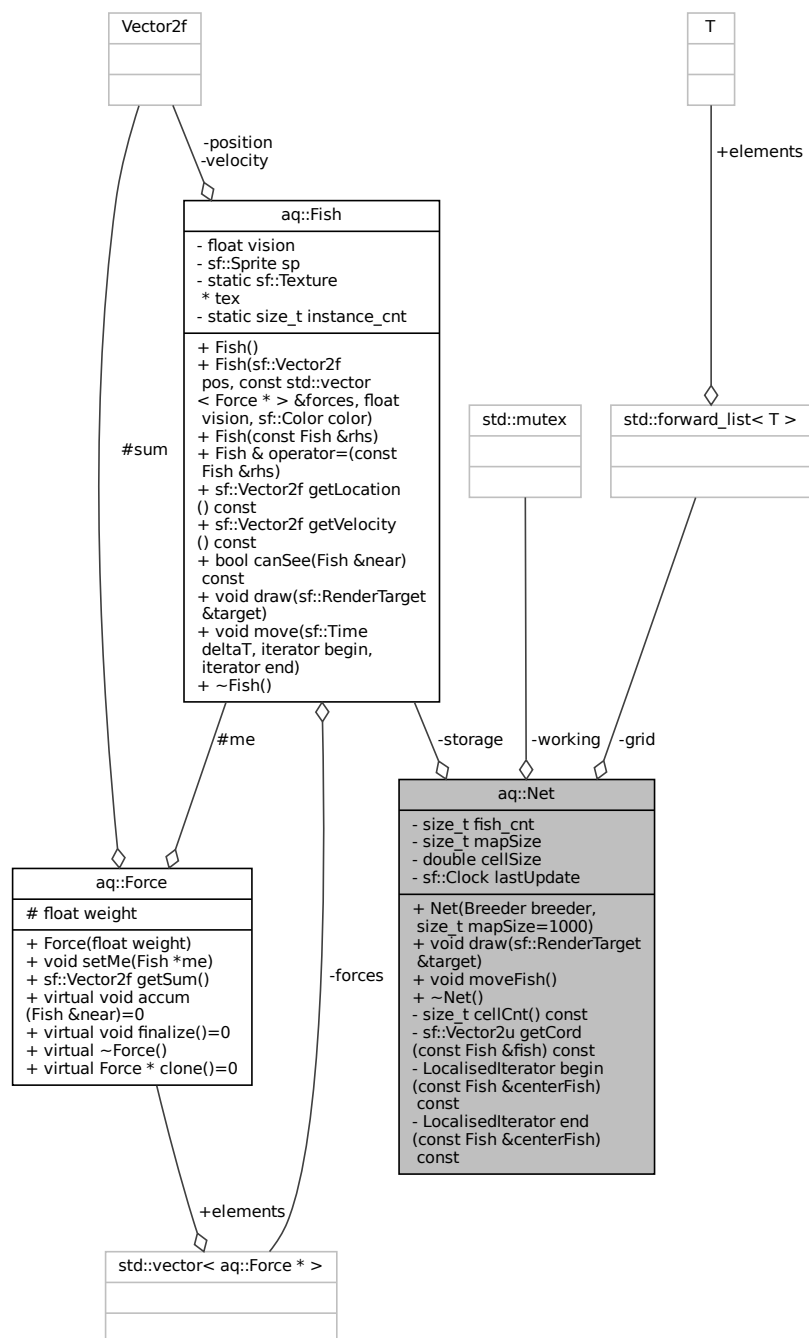
Additional Inherited Members

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

- inc/forces.hpp

3.10 aq::Net Class Reference

Collaboration diagram for aq::Net:



Classes

- class [LocalisedIterator](#)

Public Types

- typedef std::forward_list< [Fish](#) * > **cell**

Public Member Functions

- **Net** ([Breeder](#) breeder, size_t mapSize=1000)
- void **draw** (sf::RenderTarget &target)
- void **moveFish** ()

Private Member Functions

- size_t **cellCnt** () const
- sf::Vector2u **getCord** (const [Fish](#) &fish) const
- [LocalisedIterator](#) **begin** (const [Fish](#) ¢erFish) const
- [LocalisedIterator](#) **end** (const [Fish](#) ¢erFish) const

Private Attributes

- size_t **fish_cnt**
- [Fish](#) * **storage**
- cell ** **grid**
- size_t **mapSize**
- double **cellSize**
- sf::Clock **lastUpdate**
- std::mutex **working**

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

- inc/net.hpp
- src/net.cpp

3.11 shader::PerlinNoise Class Reference

Simple 2D perlin noise shader.

Collaboration diagram for shader::PerlinNoise:

shader::PerlinNoise
<ul style="list-style-type: none"> + uniform vec2 u_seed + uniform int u_octaves + uniform float u_gridSize + uniform float u_amplitude + uniform float u_water_level + uniform float u_sand_level + uniform float u_bw_mode + uniform vec4 col_low_water + uniform vec4 col_high_water + uniform vec4 col_low_sand and 6 more...
<ul style="list-style-type: none"> + float interpolate(float a, float b, float w) + float cap(float value) + vec2 randomGradient(ivec2 cord) + float dotGridGradient(ivec2 cord, vec2 pos) + float perlin(vec2 pos) + float fractalNoise(vec2 pos) + vec4 colorFromHeight(float height) + void main()

Public Member Functions

- float [interpolate](#) (float a, float b, float w)
Smoothly interpolates between two values.
- float [cap](#) (float value)
Caps a value between [0, 1].
- vec2 [randomGradient](#) (ivec2 cord)
Computes a pseudo random gradient vector for a given integer coordinate.
- float [dotGridGradient](#) (ivec2 cord, vec2 pos)
Computes the dot product of a random gradient vector and a given position.
- float [perlin](#) (vec2 pos)

2D Perlin noise

- float [fractalNoise](#) (vec2 pos)
Computes a fractal sum of perlin noise.
- vec4 [colorFromHeight](#) (float height)
Computes a color based on the height.
- void [main](#) ()
Main function.

Public Attributes

- uniform vec2 [u_seed](#)
Seed used as offset.
- uniform int [u_octaves](#)
Number of patterns to sum.
- uniform float [u_gridSize](#)
Size of the grid.
- uniform float [u_amplitude](#)
Start amplitude of the noise.
- uniform float [u_water_level](#)
Threshold for water [0, 1].
- uniform float [u_sand_level](#)
Threshold for sand [0, 1].
- uniform float [u_bw_mode](#)
B&W mask mode toggle, 0 or 1.
- uniform vec4 [col_low_water](#)
Color for deep water.
- uniform vec4 [col_high_water](#)
Color for shallow water.
- uniform vec4 [col_low_sand](#)
Color for low sand.
- uniform vec4 [col_high_sand](#)
Color for high sand.
- uniform vec4 [col_low_grass](#)
Color for low grass.
- uniform vec4 [col_high_grass](#)
Color for high grass.
- uniform vec2 [u_resolution](#)
Size of the window.
- uniform vec2 [u_top_left](#)
Top left corner of the visible area.
- uniform vec2 [u_bottom_right](#)
Bottom right corner of the visible area.

3.11.1 Detailed Description

Simple 2D perlin noise shader.

Code based on the the Perlin noise wikipedia page: https://en.wikipedia.org/wiki/Perlin_noise

Remarks

Fragment-Shader

3.11.2 Member Function Documentation

3.11.2.1 colorFromHeight()

```
vec4 shader::PerlinNoise::colorFromHeight (
    float height ) [inline]
```

Computes a color based on the height.

Parameters

<i>height</i>	in [0, 1]
---------------	-----------

3.11.2.2 fractalNoise()

```
float shader::PerlinNoise::fractalNoise (
    vec2 pos ) [inline]
```

Computes a fractal sum of perlin noise.

Returns

[0, 1]

3.11.2.3 perlin()

```
float shader::PerlinNoise::perlin (
    vec2 pos ) [inline]
```

2D Perlin noise

Parameters

<i>pos</i>	Position in 2D space
------------	----------------------

Returns

[-1, 1]

3.11.2.4 randomGradient()

```
vec2 shader::PerlinNoise::randomGradient (
    ivec2 cord ) [inline]
```

Computes a pseudo random gradient vector for a given integer coordinate.

Returns

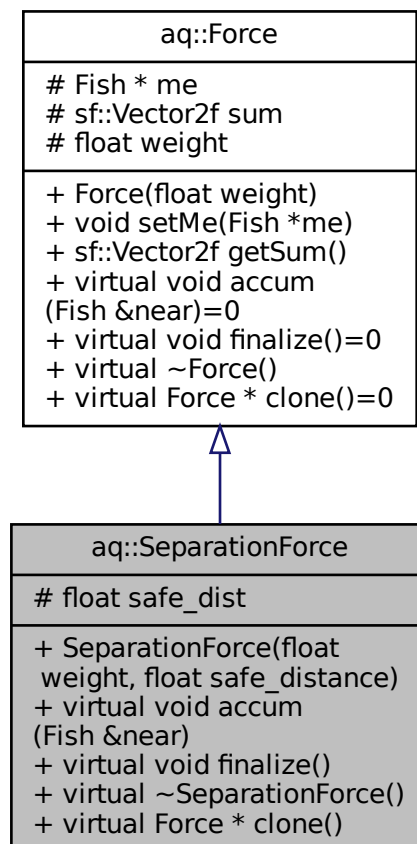
Vector with length 1

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

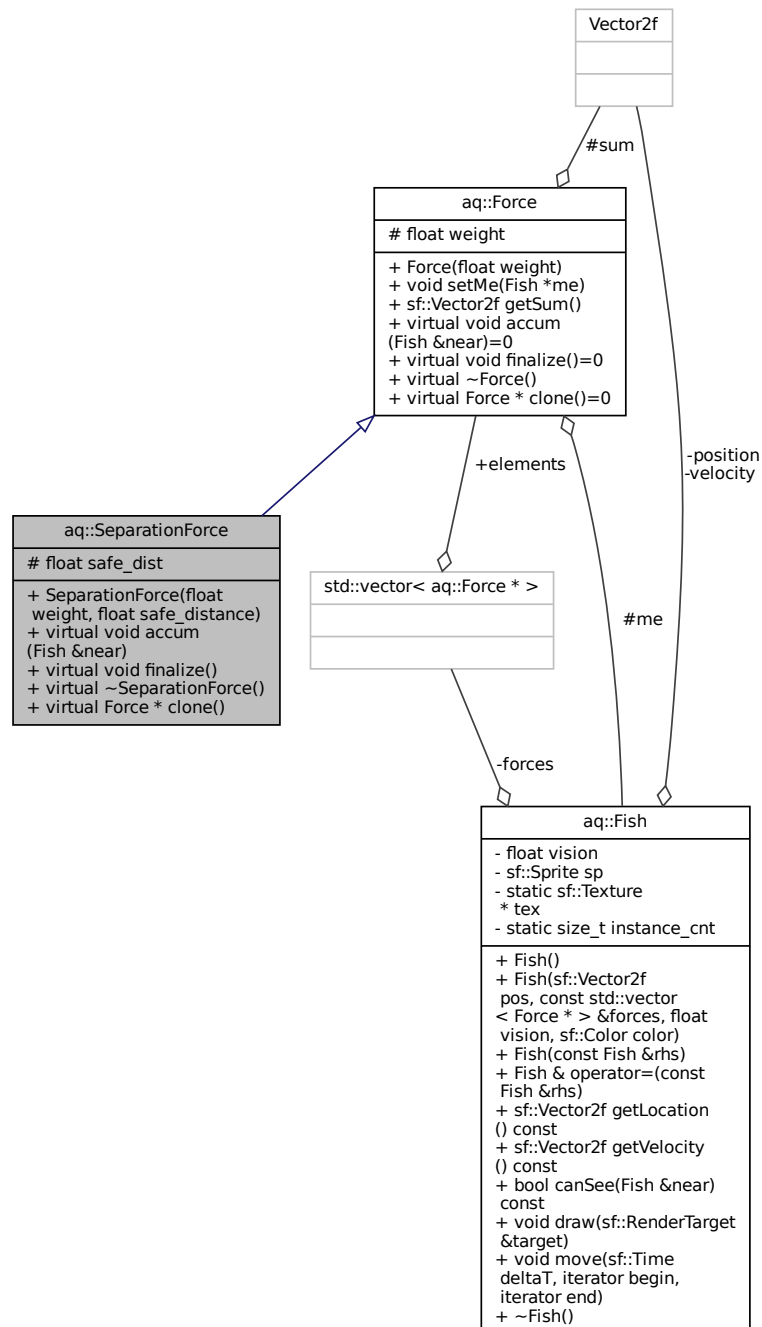
- src/perlin.frag

3.12 aq::SeparationForce Class Reference

Inheritance diagram for aq::SeparationForce:



Collaboration diagram for `aq::SeparationForce`:



Public Member Functions

- **SeparationForce** (float weight, float safe_distance)
- virtual void **accum** ([Fish](#) &near)
- virtual void **finalize** ()
- virtual [Force](#) * **clone** ()

Protected Attributes

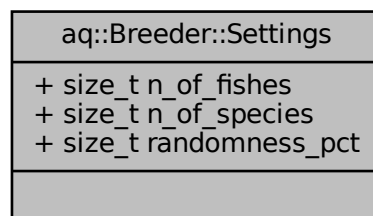
- float **safe_dist**

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

- `inc/forces.hpp`

3.13 `aq::Breeder::Settings` Struct Reference

Collaboration diagram for `aq::Breeder::Settings`:



Public Attributes

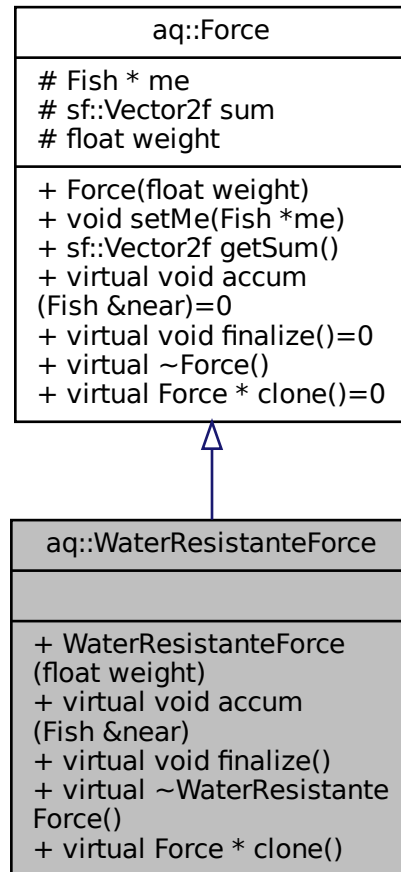
- `size_t n_of_fishes` = 100
- `size_t n_of_species` = 1
- `size_t randomness_pct` = 0

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

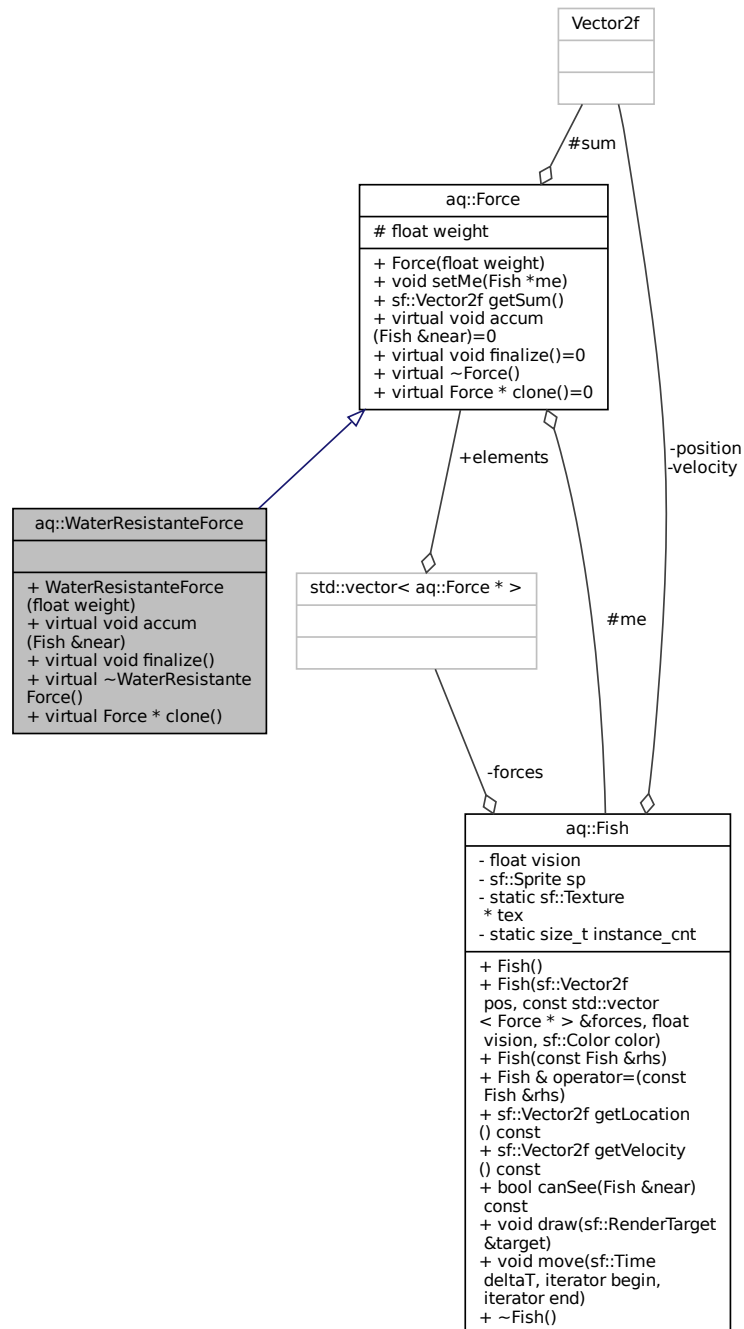
- `inc/breeder.hpp`

3.14 aq::WaterResistenteForce Class Reference

Inheritance diagram for aq::WaterResistenteForce:



Collaboration diagram for aq::WaterResistenteForce:



Public Member Functions

- **WaterResistenteForce** (float weight)
- virtual void **accum** (Fish &near)
- virtual void **finalize** ()
- virtual Force * **clone** ()

Additional Inherited Members

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

- `inc/forces.hpp`

Index

- aq::AlignmentForce, [5](#)
- aq::Breeder, [7](#)
- aq::Breeder::Settings, [27](#)
- aq::CohesionForce, [8](#)
- aq::Engine, [10](#)
- aq::Fish, [12](#)
- aq::Force, [13](#)
- aq::Island, [15](#)
- aq::MinSpeedForce, [18](#)
- aq::Net, [20](#)
- aq::Net::LocalisedIterator, [16](#)
- aq::SeparationForce, [25](#)
- aq::WaterResistantForce, [28](#)

- colorFromHeight
 - shader::PerlinNoise, [24](#)

- fractalNoise
 - shader::PerlinNoise, [24](#)

- perlin
 - shader::PerlinNoise, [24](#)

- randomGradient
 - shader::PerlinNoise, [24](#)

- shader::PerlinNoise, [22](#)
 - colorFromHeight, [24](#)
 - fractalNoise, [24](#)
 - perlin, [24](#)
 - randomGradient, [24](#)