Doc_car_dashboard

Generated by Doxygen 1.8.13

Contents

Chapter 1

README

#A Faire

chacun crée sa classe prenom_scene qui herite de la classe commune **scene_globale** (p. 136), qui elle même dérive de QGraphicsScene. Dans la **scene_globale** (p. 136) sont déclarés tout les items ci-dessous qui doivent hériter de **objet_virtuel** (p. 134). A partir du client, possibilité de changer de Dashboard par la commande CANN DASHBOARD prenom.

- un compteur Vitesse, pour value la vitesse.
- un compteur CompteTours pour value les tours par minute.
- un voyant VoyantBatterie pour value 0/1 pour éteint/allumé
- un afficheur CompteurKm, permettant d'afficher le nombre de km parcourus depuis le lancement du programme en fonction de la vitesse.
- une jauge Essence
- un voyant Clignotant avec pour value 1 le clignotant droit, -1 le clignotant gauche, et 0 éteint.
- trois voyants pour les phares : position, croisement, route. value client : 0 eteint, 1 position, 2 croisement, 3 route. Serveur : trois voyants différents.
- Warning avec pour valeur 0 eteint et 1 allumer. Met la valeur des clignotants à 2 ce qui fait que les deux clignotants s'allument.
- a completer

Chapter 2

Example

1. Lightmatter Car Dashboard - source (CC BY 1.0)
2. Rimac Concept_One Car Dashboard - source (CC BY-NC 4.0)
    - Details
3. https://i1.wp.com/autoecole-etairos.fr/wp-content/uploads/2018/03/voyants-tableau-de
   jpg?w=620&ssl=1
4. Dashboard example
5. https://i1.wp.com/autoecole-etairos.fr/wp-content/uploads/2018/03/voyants-tableau-de
   jpg?w=620&ssl=1
6. https://image.freepik.com/vecteurs-libre/tableau-bord-voiture-isole_↩
   1284-13378.jpg
7. http://www.librow.com
8. Audi A6 L e-tron Concept
9. Dashboard
10. Icon without watermark
11. https://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=imgres&cd=&cad=rja&uact=8&ved
    UKEwjyxIPi9uTlAhXx8OAKHb_nBaAQjRx6BAgBEAQ&url=https%3A%2F%2Fwww.shutterstock.↩
    com%2Fimage-photo%2Fblue-dashboard-car-meters-indicators-dark-750393586&psig=↩
    AOvVaw0AVsLEUIfsBQRdT7wuobug&ust=1573653017658449
12. The best dashboard on earth https://images-na.ssl-images-amazon.com/images/↩
    I/51XQMvX3ADL._SX355_.jpg
13. Car Dashboard UI Modern Vector
14. An example

Ford Fiesta https://cdn.autocentre.ua/ac/10/13/images/06/Fiesta_Mazda2-(11).jpg

Chapter 3

ABSProblem

- **Description:** Light that indicates ABS problem.
- **Values:** 0, 1
- **Vocal Alert:** No
- **Calculator:** No
- **Type:** LED
- **Priority:** 1

Chapter 4

AdaptiveSuspensionDampers

- **Description**: Indicator light means there is a need to contact an authorised repairer.
- **Values:**: 0, 1
- **Calculated**: Yes
- **Type**: Led
- **Priority**: 3
- **Vocal Alert**: No

Chapter 5

Airbag

- **Description:** Indicates the status of the safety airbags in cars.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 1

Chapter 6

AirSuspension

- **Description:** This is a device using an electric compressor that pumps air into a flexible bellows. This offers higher comfort and raises the chassis from the axle.
- **Values:** 0, 1
- **Vocal alert:** No
- **Calculated:** Yes
- **Type:** LED
- **Priority:** 2

Chapter 7

AllWheelDrive

- **Description:** Indicates if the car is providing power to all its wheels.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 0

Chapter 8

AutomaticGearboxWarning

- **Description**: The automatic transmission warning light means there's an issue with the transmission, possibly the fluid temperature, fluid level, or pressure.
- **Values**: 0, 1
- **Calculated** No
- **Type**: LED
- **Priority**: 2

Chapter 9

AutomaticTransmissionModes

- **Description:** 8 Indicator lights representing the modes of the automatic driving : Park (P), Reverse (R), Drive (D), Neutral (N), First (1 or L [Low]), Second (2 or S), Third (3), Overdrive (D or OD).
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 0

Chapter 10

BonnetOpenWarningLight

- **Description:** Your bonnet might not be fully closed. If the bonnet is completely secure and the warning light is still on then contact a repair service to fix this issue.
- **Values:** 0, 1
- **Calculated:** No
- **Type:** LED
- **Priority:** 2
- **Vocal alert:** Yes

Chapter 11

BootLidWarningLight

- **Description:** Your boot might not be fully closed. If the boot is completely secure and the warning light is still on then contact a repair service to fix this issue.
- **Values::** 0, 1
- **Calculated:** No
- **Type:** Led
- **Priority:** 2
- **Vocal alert:** Yes

Chapter 12

BrakeFluid

- **Description:** LED that is turned on when the remaining volume of the brake fluid is critical (below a certain value).
- **Values:** 0,1
- **Vocal alert:** Yes
- **Calculator:** No
- **Type:** LED
- **Priority:** 1

Chapter 13

BrakePads

- **Description:** LED that is turned on when the brake pads are damaged.
- **Values:** 0, 1
- **Vocal alert:** Yes
- **Calculator:** No
- **Type:** LED
- **Priority:** 1

Chapter 14

Camera

- **Description:** A camera on the front, back, dashboard or side of the car that records and display the view.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** Video
- **Priority:** 0

Chapter 15

CheckEngineLight

- **Description:** Indicator light that turns on whenever the engine is turned on to check the bulb. If the light stays illuminated, the car's diagnostic systems have detected a malfunction that needs to be investigated.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 1-2

Chapter 16

ClutchPedal

- **Description:** LED that is turned on when the clutch pedal is pressed.
- **Values:** 0, 1
- **Vocal alert:** No
- **Calculator:** No
- **Type:** LED
- **Priority:** 0

Chapter 17

CoolantTemperature

- **Description:** Here is a temperature guage that indicates if the engine's coolant temperature is *cold*, *normal* or *overheating*.
- **Values:** 160℃ to 200℃
- **Vocal alert:** Yes
- **Calculated:** Yes
- **Automatic:** Yes
- **Type:** Gauge 60 degrees, LED
- **Priority:** 3

Chapter 18

CruiseControlIsActivated

- **Description:** Indicator light that turns on when the Cruise control is activated by the driver. The cruise control maintains a steady speed as set by the driver.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 1

Chapter 19

ESPFault

- **Description**: Indicator light that means that there is a problem with the vehicle's traction control.
- **Values:**: 0, 1
- **Calculated**: No
- **Type**: LED
- **Priority**: 2
- **Vocal alert**: No

Chapter 20

FaultyBulb

- **Description:** If any of your bulbs are faulty or blown, this light will warn you that one of the exterior bulbs has an issue. Most modern vehicles will tell you which bulb it is.
- **Values:**
  - **0:** No bulb failure, the LED is off.
  - **1:** There is an issue with the headlamp leveling or AFS, the LED is flashing.
  - **2:** There is a light bulb failure, the LED is on/solid.
- **Calculated:** No
- **Type:** Led
- **Priority:** 1
- **Vocal Alert:** No

Chapter 21

FourWDLock

- **Description**: Indicator light that means that the vehicle's 4WD Lock mode is activated.
- **Values:**: 0, 1
- **Calculated**: Yes
- **Type**: LED
- **Priority**: 0
- **Vocal Alert**: No

Chapter 22

FrontAntifog

- **Description:** The front antifog lights reveal the position of the vehicle when driving in the fog, snow or other conditions that limit visibility. This helps make the vehicle visible to oncoming traffic. The rear lights illuminate when you push the button.
- **Values:** 0, 1
- **Priority:** 2-3
- **Vocal alert:** No
- **Calculated:** No
- **Type:** LED

Chapter 23

FuelGauge

- **Description:** Gauge that indicates the remaining quantity of fuel in the tank.
- **Values:** 0-1 (0 empty, 1 full)
- **Vocal alert:** No
- **Calculator:** Yes
- **Type:** Gauge 120
- **Priority:** 1

Chapter 24

Handbrake

- **Description:** An indicator light turns on when the handbrake is on.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 2

Chapter 25

HazardLights

- **Description:** Blinking LED that is turned on by the driver to signal a sudden danger to other drivers.
- **Values:** 0, 1
- **Vocal alert:** No
- **Calculator:** No
- **Type:** Blinking LED
- **Priority:** 1

Chapter 26

HighbeamHeadlights

- **Description:** When the high beam headlights are on, the function turns on the corresponding LED symbol.
- **Values:** 0, 1
- **Priority:** 1
- **Vocal alert :** No
- **Calculated:** No
- **Type:** LED
- **Comments:** When the value is 1 (ON) the corresponding values of Low-Beam HeadLights and Day-Time Headlights should be 0 (OFF). The value of High-beam Headlights is independent of Antifog light.

Chapter 27

LightSensor

- **Description:** Turn lights on or off automatically depending on the intensity of the light outside.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 1

Chapter 28

LowBattery

- **Description:** Indicator light means that the car's charging system is short of power or is not charging properly. It normally indicates a problem with the battery itself or the alternator.
- **Values:** 0, 1
- **Calculated:** No
- **Type:** LED
- **Priority:** 3
- **Vocal alert:** No

Chapter 29

LowFuelLevel

- **Description:** Indicator light that indicates that the car is running low on fuel and will soon need a refill.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 3

Chapter 30

MalfunctionOfThePollutionControlSystem

- **Description:** Indicator light that turns on when there is a malfunction in the system employed to limit the discharge of noxious gases from the internal- combustion engine and other components.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 2

Chapter 31

MotorTemperatureGauge

- **Description:** Gauge that indicates the temperature of the motor.
- **Values:** 0-3
- **Vocal alert:** No
- **Calculator:** Yes
- **Type:** Gauge 120
- **Priority:** 1

Chapter 32

OilLight

- **Description:** Light that indicates the low level of pressure for oils.
- **Values:** 0, 1
- **Vocal Alert:** No
- **Calculator:** No
- **Type:** LED
- **Priority:** 0

Chapter 33

ParkingAssistant

- **Description:** Helps the driver to park by detecting objects. It uses different sonor intensity.
- **Value:** 0, 1
- **Vocal Alert:** Yes
- **Calculated:** Yes
- **Type:** Sound
- **Priority:** 2

Chapter 34

PowerSteeringWarningLight

- **Description:** There may be an issue with your power steering. Remove the key from the ignition for 30 seconds and start the engine again. If the light still appears, contact a garage as soon as possible.
- **Values::** 0, 1
- **Calculated:** No
- **Type:** LED
- **Priority:** 3
- **Vocal alert:** No

Chapter 35

PreHeatingDiesel

- **Description:** Light that indicates if the pre heating is okay.
- **Values:** 0, 1
- **Vocal Alert:** No
- **Calculator:** No
- **Type:** LED
- **Priority:** 1

Chapter 36

RainSensor

- **Description:** A sensor that activates the windscreen wiper when it rains.
- **Values:** 0, 1
- **Vocal Alert:** No
- **Calculator:** No
- **Type:** Sensor
- **Priority:** 1

Chapter 37

RearAntifogLight

- **Description:** Rear antifog lights reveal the position of the vehicle when driving in the fog, snow or other conditions that limit visibility. Rear lights illuminate when you push the button.
- **Values:** 0, 1
- **Priority:** 2-3
- **Vocal alert :** No
- **Calculated:** No
- **Type:** LED
- **Comments:** Rear fog lights are mandatory in Europe.ml

Chapter 38

RearWindowHeating

- **Description:** System to clear condensation and thaw frost from the rear window of a car.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 0

Chapter 39

Seatbelt

- **Descriptor:** Indicates if the seatbelt is properly attached.
- **Values:** 0, 1
- **Calculated:** No
- **Type:** LED
- **Priority:** 1

Chapter 40

SecurityLight

- **Description:** The security light is a visual indication that indicates something in the car's antitheft system is failing.
- **Values:** 0, 1
- **Calculated:** No
- **Type:** Led
- **Priority:** 3

Chapter 41

SecuritySystemEnabled

- **Description:** The indicator light blinks if the car is locked and the security system is enabled. It will need the proper key to deactivate.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** No
- **Type:** LED
- **Priority:** 1

Chapter 42

ServiceVehicleSoon

- **Description:** This indicator light turns on when there is a faulty condition in an area of the vehicle chassis systems such as the anti-lock brake system (ABS), the traction control system (TCS), the electronic suspension system, or the brake hydraulic system.
- **Values::** 0, 1
- **Calculated:** Yes
- **Type:** LED
- **Priority:** 3
- **Vocal Alert:** No

Chapter 43

Speedometer

- **Description:** Gauge that indicates the speed of the car.
- **Values:** 0 to 320
- **Vocal Alert:** No
- **Calculator:** No
- **Type:** Gauge
- **priority:** 3

Chapter 44

SteeringWheelLock

- **Description:** If this warning light appears, do not drive your vehicle. There may be an issue with your steering lock. The indicator light means your steering wheel is locked and can not be moved. To turn off the steering lock, insert the key into the ignition and turn it to at least the first position while turning the steering wheel in either direction.
- **Values:** 0, 1
- **Calculated:** No
- **Type:** Led
- **Priority:** 3 (Safety)
- **Vocal alert:** No

Chapter 45

TachometerGauge

- **Description:** The tachometer measures the rotation speed of the car in revolutions per minute (rpm), displayed on the dashboard in (X 1000). In practical terms, it measures the rotation speed of the engine's crankshaft.
- **Values:** 0-8
- **Vocal alert:** No
- **Calculated:** No
- **Automatic:** Yes
- **Type:** LED, Circular Gauge (180°)
- **Priority:** 1

Chapter 46

TirePressure

- **Description:** LED that is turned on when the pressure of a tire is below a certain value.
- **Values:** 0, 1
- **Vocal alert:** Yes
- **Calculator:** No
- **Type:** LED
- **Priority:** 2

Chapter 47

TurnSignal

- **Description:** LED turned on when the left or right signal is activated.
- **Values:** 0 or 1
- **Vocal alert:** No
- **Calculator:** No
- **Type:** LED
- **Priority:** 2

Chapter 48

UnclosedDoor

- **Description:** An indicator light and a vocal signal turn on when at least one door is not shut properly.
- **Values:** 0, 1
- **Calculated:** No
- **Vocal alert:** Yes
- **Type:** LED
- **Priority:** 2

Chapter 49

WasherFluidIndicator

- **Description**: This light's only function is to let the driver know the washer fluid is low and to remind them to go fill it up.
- **Values**: 0, 1
- **Calculated**: No
- **Type**: Led
- **Priority**: 0

Chapter 50

Hierarchical Index

50.1 Class Hierarchy

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

Chapter 51

Class Index

51.1 Class List

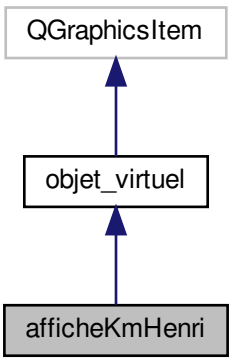
Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 52

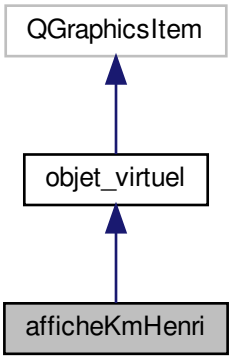
Class Documentation

52.1 afficheKmHenri Class Reference

Inheritance diagram for afficheKmHenri:



Collaboration diagram for afficheKmHenri:



Public Member Functions

- **afficheKmHenri** (**objet_virtuel** ∗parent=nullptr)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

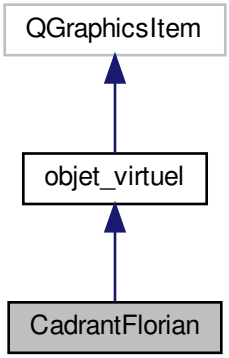
Additional Inherited Members

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

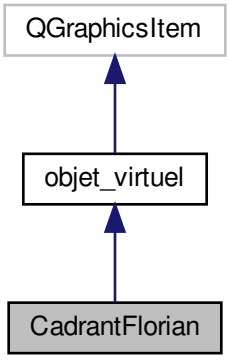
- serveur/Henri/affichekmhenri.h
- serveur/Henri/affichekmhenri.cpp

52.2 CadrantFlorian Class Reference

Inheritance diagram for CadrantFlorian:



Collaboration diagram for CadrantFlorian:



Public Member Functions

- **CadrantFlorian** (bool hasText=true, bool hasSubTrait=true, int invertAiguille=1, int pas=10, int angleB=220, int angleE=260, int valeurMax=260, QGraphicsItem ∗parent=NULL)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- float **generateAngle** ()
- int **getSpeedMax** ()

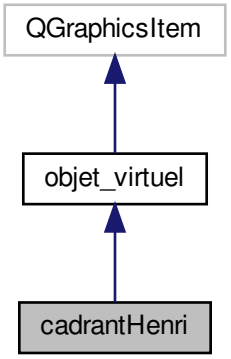
Additional Inherited Members

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

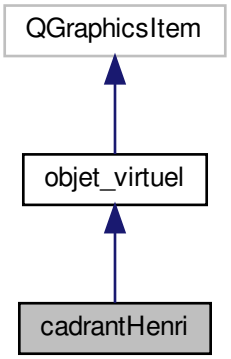
- serveur/Florian/cadrantflorian.h
- serveur/Florian/cadrantflorian.cpp

52.3 cadrantHenri Class Reference

Inheritance diagram for cadrantHenri:



Collaboration diagram for cadrantHenri:



Public Member Functions

- **cadrantHenri** (**objet_virtuel** ∗parent=nullptr)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

Public Attributes

- int **epesseurTraitVitesse**
- int **tailleTexteVitesse**

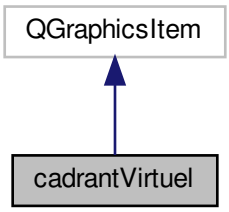
Additional Inherited Members

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

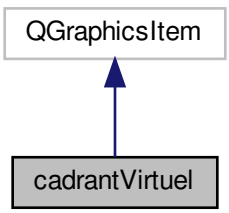
- serveur/Henri/cadranthenri.h
- serveur/Henri/cadranthenri.cpp

52.4 cadrantVirtuel Class Reference

Inheritance diagram for cadrantVirtuel:



Collaboration diagram for cadrantVirtuel:



Public Member Functions

- **cadrantVirtuel** (QGraphicsItem ∗parent=nullptr)
- int **getValeur** () const
- void **setValeur** (int value)
- int **getValeurMax** () const

Protected Attributes

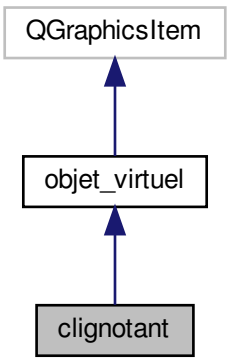
- int **valeur**
- int **valeurMax**

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

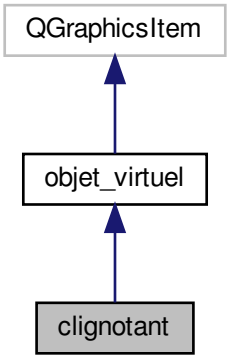
- serveur/Florian/cadrantvirtuel.h
- serveur/Florian/cadrantvirtuel.cpp

52.5 clignotant Class Reference

Inheritance diagram for clignotant:



Collaboration diagram for clignotant:



Public Member Functions

- **clignotant** (QGraphicsItem ∗parent=nullptr)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- void **clignoter** ()

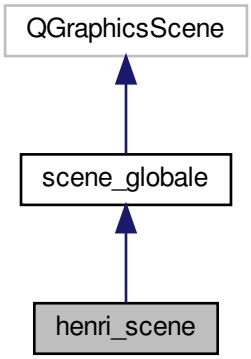
Additional Inherited Members

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

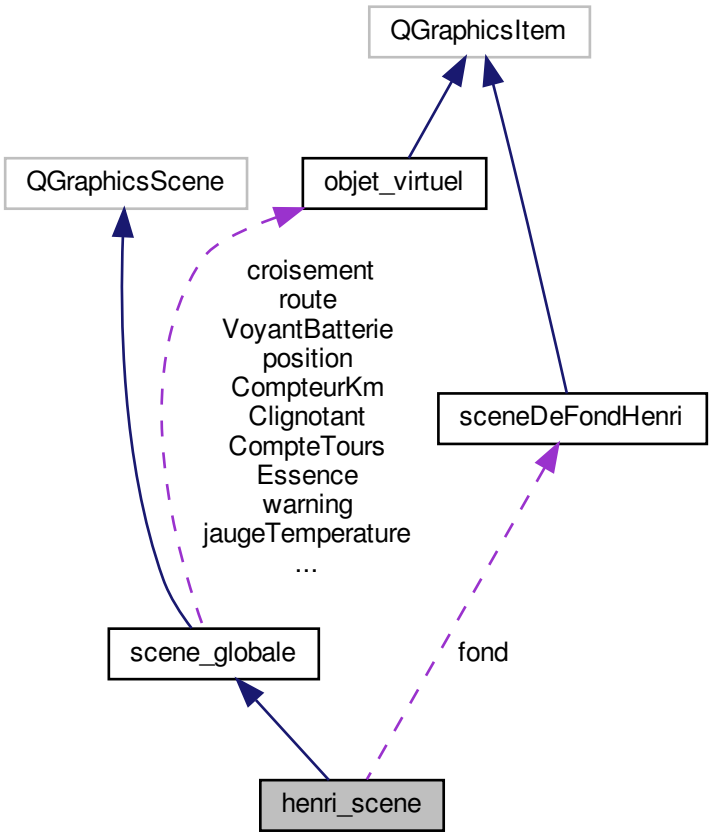
- serveur/Florian/clignotant.h
- serveur/Florian/clignotant.cpp

52.6 henri_scene Class Reference

Inheritance diagram for henri_scene:



Collaboration diagram for henri_scene:



Public Member Functions

- **henri_scene** (**scene_globale** ∗parent=nullptr)
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const

Public Attributes

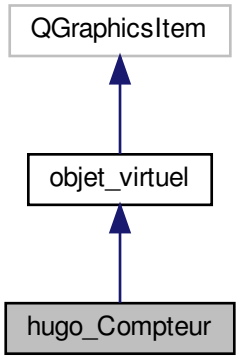
- **sceneDeFondHenri** ∗ **fond**

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

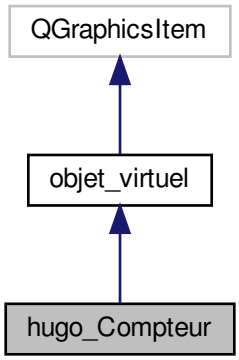
- serveur/Henri/henri_scene.h
- serveur/Henri/henri_scene.cpp

52.7 hugo_Compteur Class Reference

Inheritance diagram for hugo_Compteur:



Collaboration diagram for hugo_Compteur:



Public Member Functions

- **hugo_Compteur** (int, int, int, int, int, QStringList, int, int, int, int, int, int, int param_critique=100, int red2=100, int green2=100, int blue2=100)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- void **Parametrage** (int param_x, int param_y, int param_r, int param_start_angle, int param_end_angle, QStringList param_graduations, int param_value, int param_r_verre, int param_direction_grad, int red, int green, int blue, int param_critique=100, int red2=100, int green2=100, int blue2=100)

Protected Attributes

- int **x** =0
- int **y** =0
- int **r** =100
- int **start_angle** =0
- int **end_angle** =360
- int **nbre_graduations** =12
- int **critique**
- int **angle** =0
- int **value2** =0
- int **r_verre**
- int **direction_grad** =1
- QColor **couleur** =QColor(100,100,100)
- QColor **couleur2** =QColor(100,100,100)
- QColor **couleurgrad** =QColor(100,100,100,50)
- QColor **couleurgrad2** =QColor(100,100,100,50)
- QStringList **graduations**

Additional Inherited Members

52.7.1 Member Function Documentation

52.7.1.1 paint()

```
void hugo_Compteur::paint (
            QPainter * painter,
            const QStyleOptionGraphicsItem * option,
            QWidget * widget )
```

<Mise en place Antialiasing//

La première étape est la création de l'arc de cercle servant de support au cadran. L'option critique permet d'ajouter une autre couleur à une partie du cadran

Création des graduations en deux boucles (grandes et petites graduations)

Ajout du texte sur les graduations

Mise en place de l'aiguille

<Cercle au centre de l'aiguille//

<Dégradé à la base de l'aiguille pour donner une impression de relief//

Création d'un polygone pour représenter l'aiguille, et affichage

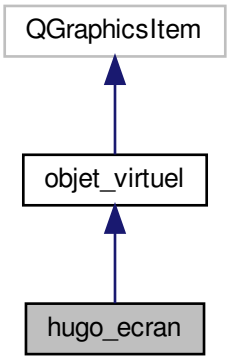
Affichage d'un effet verre en superposant un cercle transparent avec gradient blanc->noir sur le cadran//

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

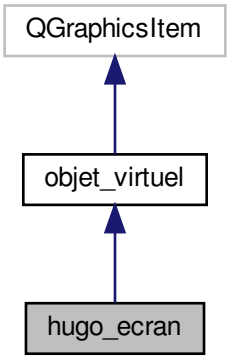
- serveur/Hugo/hugo_compteur.h
- serveur/Hugo/hugo_compteur.cpp

52.8 hugo_ecran Class Reference

Inheritance diagram for hugo_ecran:



Collaboration diagram for hugo_ecran:



Public Member Functions

- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

Additional Inherited Members

52.8.1 Member Function Documentation

52.8.1.1 paint()

```
void hugo_ecran::paint (
            QPainter * painter,
            const QStyleOptionGraphicsItem * option,
            QWidget * widget )
```

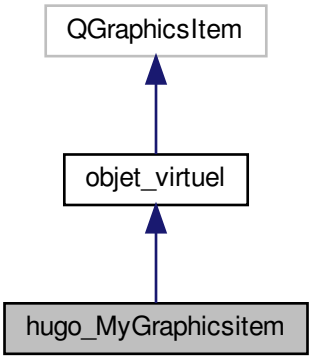
< Conversion pour avoir l'affichage sous forme heure.minutes

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

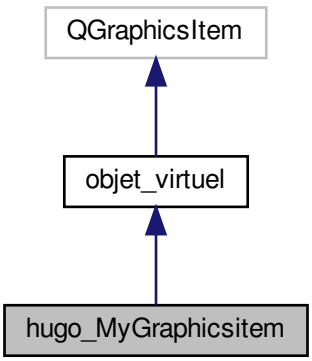
- serveur/Hugo/hugo_ecran.h
- serveur/Hugo/hugo_ecran.cpp

52.9 hugo_MyGraphicsitem Class Reference

Inheritance diagram for hugo_MyGraphicsitem:



Collaboration diagram for hugo_MyGraphicsitem:



Public Member Functions

- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

Public Attributes

- int **current_speed** =30
- QString **station** =""
- float **km** =0.0

Additional Inherited Members

52.9.1 Member Function Documentation

52.9.1.1 paint()

```
void hugo_MyGraphicsitem::paint (
            QPainter * painter,
            const QStyleOptionGraphicsItem * option,
            QWidget * widget )
```

<Antialiasing//

Création du fond (gradient ou image)

Création des arcs de cercles gris, et d'un fond noir pour poser les objects

Création de l'arc de cercle fermé bleu qui entoure le compteur de vitesse

Affichage de texte dans les différents compteurs

Création d'un afficheur. Affiche le temps réel, aisi que la distance parcourue depuis le départ

Afficheur pour les stations de radios

Affichage régime moteur

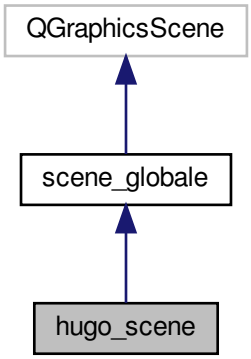
Affichage d'une icone jauge d'essence if (getValue()==1){///< Va chercher la valeur de la variable "valeur" de la classe mère. Si ==1, le voyant doit être affiché

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

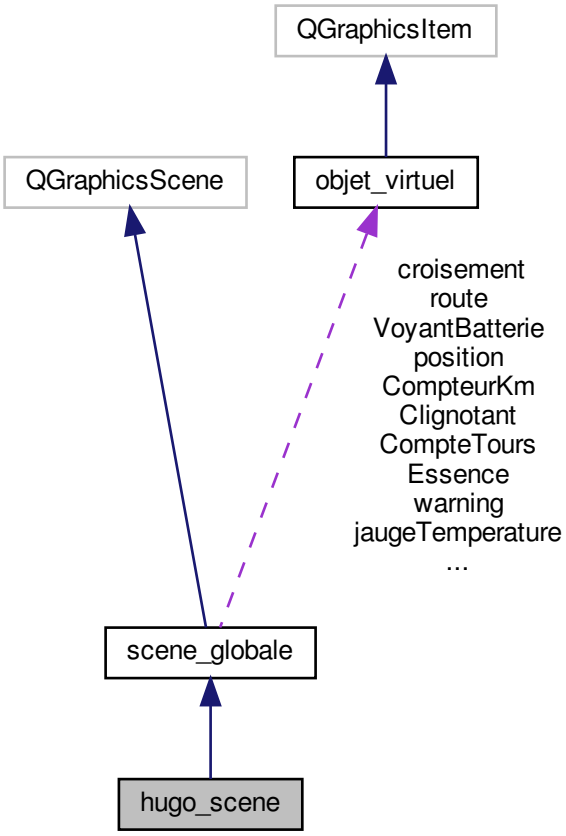
- serveur/Hugo/hugo_mygraphicsitem.h
- serveur/Hugo/hugo_mygraphicsitem.cpp

52.10 hugo_scene Class Reference

Inheritance diagram for hugo_scene:



Collaboration diagram for hugo_scene:



Public Member Functions

- **hugo_scene** (**scene_globale** ∗parent=nullptr)

Additional Inherited Members

52.10.1 Constructor & Destructor Documentation

52.10.1.1 hugo_scene()

```
hugo_scene::hugo_scene (
            scene_globale * parent = nullptr )
```

Création de la scène de fond

Déclaration et paramétrage de tous les compteurs

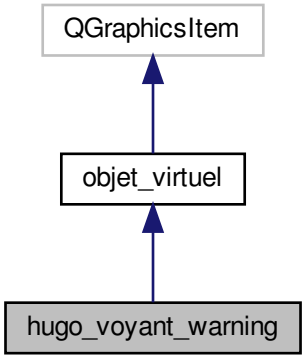
Déclaration et paramétrages de tous les voyants simples

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

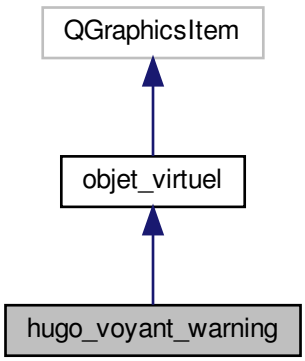
- serveur/Hugo/hugo_scene.h
- serveur/Hugo/hugo_scene.cpp

52.11 hugo_voyant_warning Class Reference

Inheritance diagram for hugo_voyant_warning:



Collaboration diagram for hugo_voyant_warning:



Public Member Functions

- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const
- void **MAJ** ()

  hugo_voyants_clignotant::MAJ *(*p. *121) Fonction de mise à jour de l'affichage. La valeur cligno controle l'opacité du painter, permettant de le rendre visible ou non*

Public Attributes

- int **cligno**

Additional Inherited Members

52.11.1 Member Function Documentation

52.11.1.1 paint()

```
void hugo_voyant_warning::paint (
            QPainter * painter,
            const QStyleOptionGraphicsItem * option,
            QWidget * widget )
```

<Antialiasing//

Va chercher la valeur de la variable "valeur" de la classe hugo_voyants. 1 correspond à l'affichaqge du clignotant droit, -1 à celui du clignotant gauche.

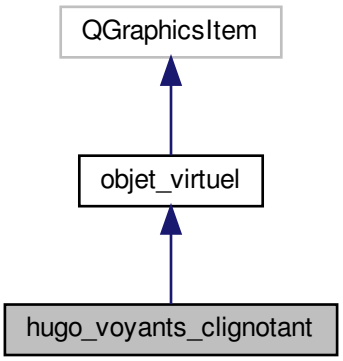
<La mise à jour est effectuée à l'aide d'un timer, permettant un affichage alterné

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

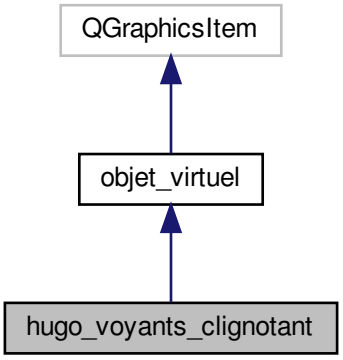
- serveur/Hugo/hugo_voyant_warning.h
- serveur/Hugo/hugo_voyant_warning.cpp

52.12 hugo_voyants_clignotant Class Reference

Inheritance diagram for hugo_voyants_clignotant:



Collaboration diagram for hugo_voyants_clignotant:



Public Member Functions

- **hugo_voyants_clignotant** ()

  hugo_voyants_clignotant::hugo_voyants_clignotant *(*p. *122) Constructeur avec initialisation des paramètres de la classe*
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const
- void **MAJ** ()

  hugo_voyants_clignotant::MAJ *(*p. *121) Fonction de mise à jour de l'affichage. La valeur cligno controle l'opacité du painter, permettant de le rendre visible ou non*

Public Attributes

- int **cligno**

Additional Inherited Members

52.12.1 Constructor & Destructor Documentation

52.12.1.1 hugo_voyants_clignotant()

```
hugo_voyants_clignotant::hugo_voyants_clignotant ( )
```

hugo_voyants_clignotant::hugo_voyants_clignotant (p. 122) Constructeur avec initialisation des paramètres de la classe

Classe permettant d'afficher les voyants pouvant clignoter

52.12.2 Member Function Documentation

52.12.2.1 paint()

```
void hugo_voyants_clignotant::paint (
            QPainter * painter,
            const QStyleOptionGraphicsItem * option,
            QWidget * widget )
```

<Antialiasing//

Va chercher la valeur de la variable "valeur" de la classe hugo_voyants. 1 correspond à l'affichaqge du clignotant droit, -1 à celui du clignotant gauche.

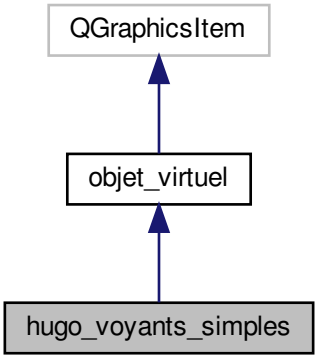
<La mise à jour est effectuée à l'aide d'un timer, permettant un affichage alterné

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

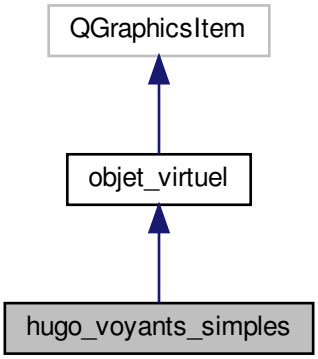
- serveur/Hugo/hugo_voyants_clignotant.h
- serveur/Hugo/hugo_voyants_clignotant.cpp

52.13 hugo_voyants_simples Class Reference

Inheritance diagram for hugo_voyants_simples:



Collaboration diagram for hugo_voyants_simples:



Public Member Functions

- **hugo_voyants_simples** (int, int, QString, int red=255, int green=0, int blue=0, int param_size=30)

  hugo_voyants_simples::hugo_voyants_simples *(*p. *124). Constructeur avec initialisation des paramètres de la classe.*
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const

Protected Attributes

- int **x**
- int **y**
- int **size** =30
- QString **chemin**
- QColor **couleur**

Additional Inherited Members

52.13.1 Constructor & Destructor Documentation

52.13.1.1 hugo_voyants_simples()

```
hugo_voyants_simples::hugo_voyants_simples (
            int param_x,
            int param_y,
            QString param_chemin,
            int red = 255,
            int green = 0,
            int blue = 0,
            int param_size = 30 )
```

hugo_voyants_simples::hugo_voyants_simples (p. 124). Constructeur avec initialisation des paramètres de la classe.

Classe permettant l'affichage de voyants simples i.e. dont l'état est allumé ou éteint

52.13.2 Member Function Documentation

52.13.2.1 paint()

```
void hugo_voyants_simples::paint (
            QPainter * painter,
            const QStyleOptionGraphicsItem * option,
            QWidget * widget )
```

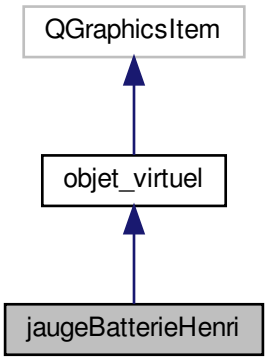
< Va chercher la valeur de la variable "valeur" de la classe mère. Si ==1, le voyant doit être affiché

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

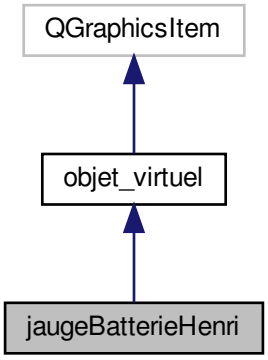
- serveur/Hugo/hugo_voyants_simples.h
- serveur/Hugo/hugo_voyants_simples.cpp

52.14 jaugeBatterieHenri Class Reference

Inheritance diagram for jaugeBatterieHenri:



Collaboration diagram for jaugeBatterieHenri:



Public Member Functions

- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const

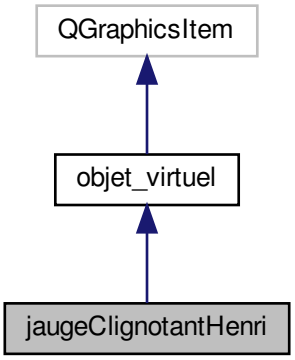
Additional Inherited Members

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

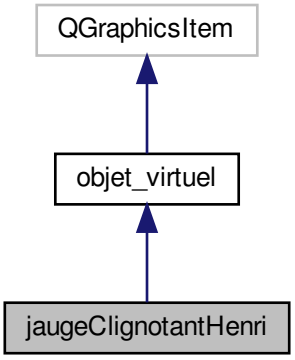
- serveur/Henri/jaugebatteriehenri.h
- serveur/Henri/jaugebatteriehenri.cpp

52.15 jaugeClignotantHenri Class Reference

Inheritance diagram for jaugeClignotantHenri:



Collaboration diagram for jaugeClignotantHenri:



Public Member Functions

- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const
- void **MAJ** ()

Public Attributes

- int **cligno**

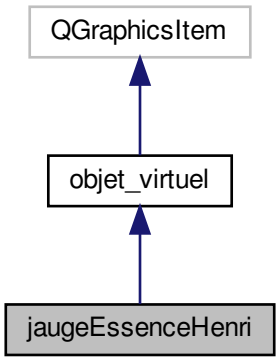
Additional Inherited Members

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

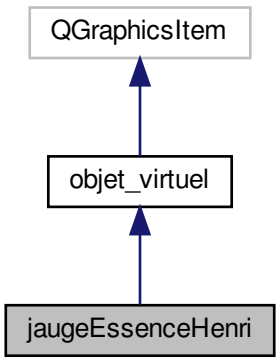
- serveur/Henri/jaugeclignotanthenri.h
- serveur/Henri/jaugeclignotanthenri.cpp

52.16 jaugeEssenceHenri Class Reference

Inheritance diagram for jaugeEssenceHenri:



Collaboration diagram for jaugeEssenceHenri:



Public Member Functions

- **jaugeEssenceHenri** (**objet_virtuel** ∗parent=nullptr)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

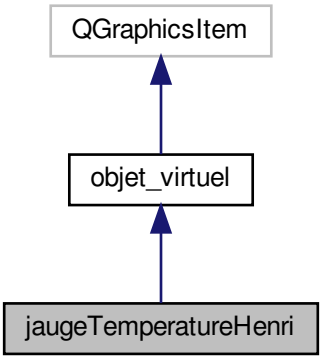
Additional Inherited Members

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

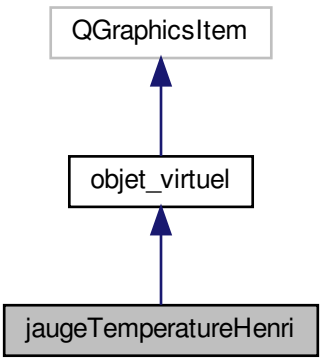
- serveur/Henri/jaugeessencehenri.h
- serveur/Henri/jaugeessencehenri.cpp

52.17 jaugeTemperatureHenri Class Reference

Inheritance diagram for jaugeTemperatureHenri:



Collaboration diagram for jaugeTemperatureHenri:



Public Member Functions

- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

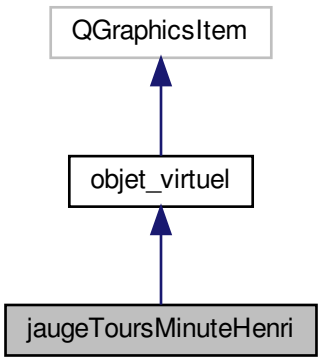
Additional Inherited Members

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

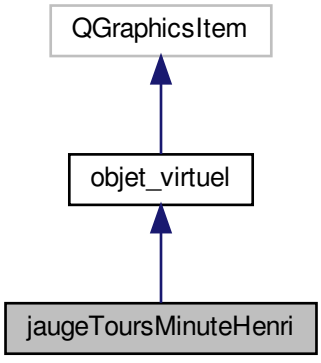
- serveur/Henri/jaugetemperaturehenri.h
- serveur/Henri/jaugetemperaturehenri.cpp

52.18 jaugeToursMinuteHenri Class Reference

Inheritance diagram for jaugeToursMinuteHenri:



Collaboration diagram for jaugeToursMinuteHenri:



Public Member Functions

- **jaugeToursMinuteHenri** (**objet_virtuel** ∗parent=nullptr)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

Public Attributes

- int **epesseurTraitToursMinure**
- int **tailleTextetoursMinute**
- QString **styleTexte**

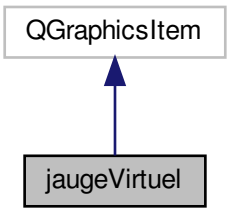
Additional Inherited Members

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

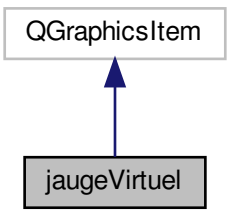
- serveur/Henri/jaugetoursminutehenri.h
- serveur/Henri/jaugetoursminutehenri.cpp

52.19 jaugeVirtuel Class Reference

Inheritance diagram for jaugeVirtuel:



Collaboration diagram for jaugeVirtuel:



Public Member Functions

- **jaugeVirtuel** (QGraphicsItem ∗parent=nullptr)
- int **getValeur** () const
- void **setValeur** (int value)
- int **getValeurMax** () const

Protected Attributes

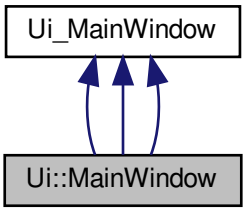
- int **valeur**
- int **valeurMax**

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

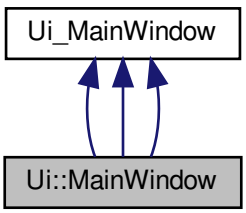
- serveur/Florian/jaugevirtuel.h
- serveur/Florian/jaugevirtuel.cpp

52.20 Ui::MainWindow Class Reference

Inheritance diagram for Ui::MainWindow:



Collaboration diagram for Ui::MainWindow:



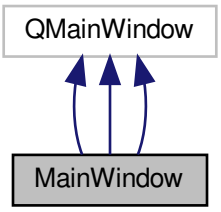
Additional Inherited Members

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

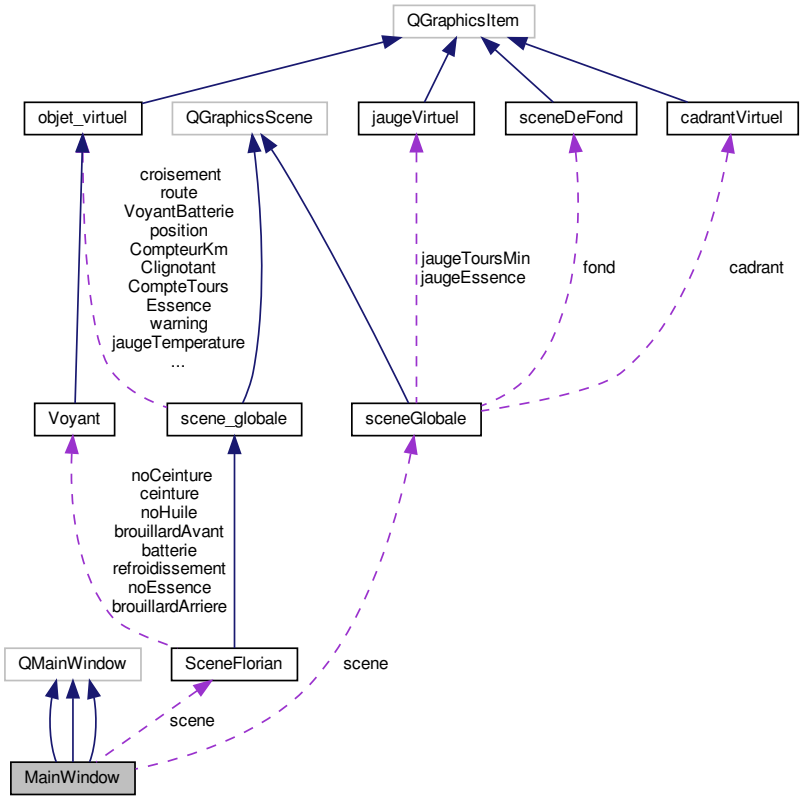
- build-serveur-Desktop-Debug/ui_mainwindow.h

52.21 MainWindow Class Reference

Inheritance diagram for MainWindow:



Collaboration diagram for MainWindow:



Public Member Functions

- **MainWindow** (QWidget ∗parent=0)
- **MainWindow** (QWidget ∗parent=0)
- void **cli** ()
- **MainWindow** (QWidget ∗parent=0)

Public Attributes

- **SceneFlorian** ∗ **scene**
- **sceneGlobale** ∗ **scene**

52.21.1 Constructor & Destructor Documentation

52.21.1.1 MainWindow()

```
MainWindow::MainWindow (
            QWidget * parent = 0 )  [explicit]
```

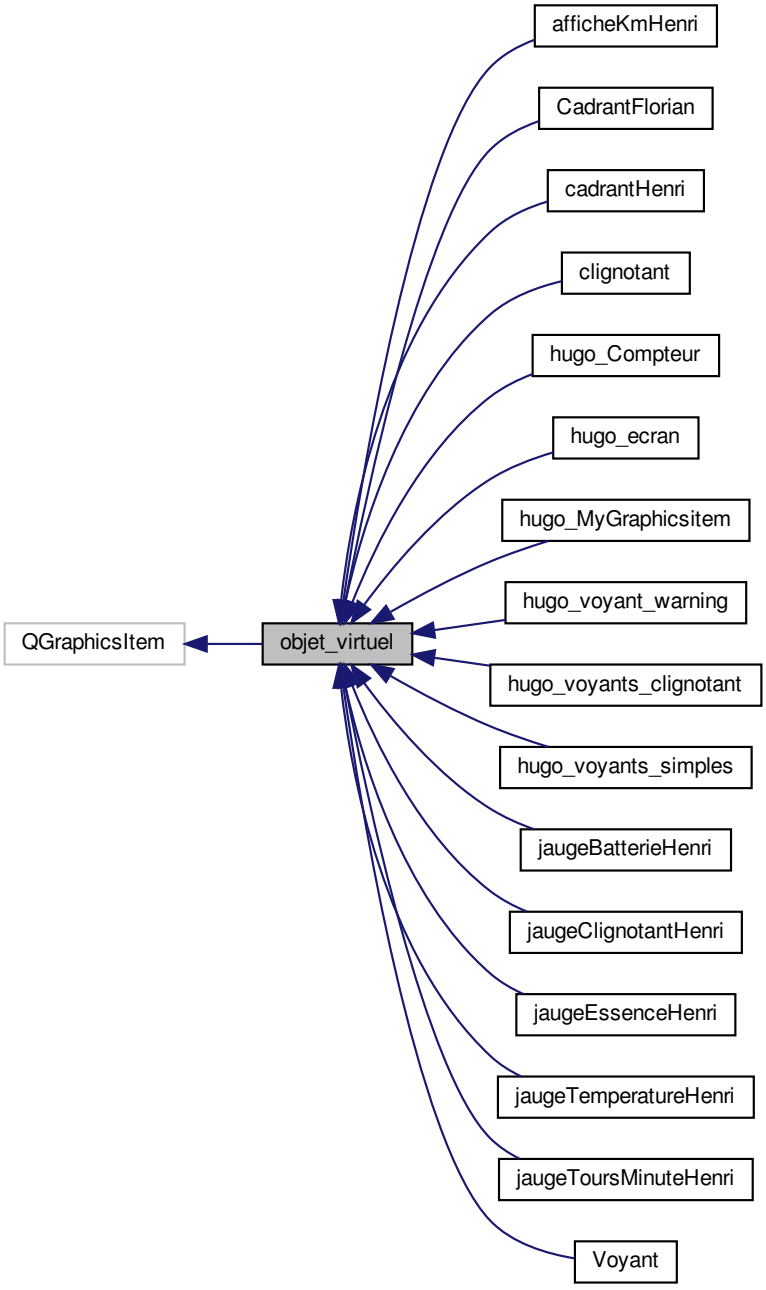
La scène par défault est

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

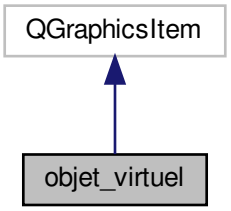
- serveur/Florian/mainwindow.h
- serveur/Florian/mainwindow.cpp

52.22 objet_virtuel Class Reference

Inheritance diagram for objet_virtuel:



Collaboration diagram for objet_virtuel:



Public Member Functions

- **objet_virtuel** (QGraphicsItem ∗parent=nullptr)
- int **getValue** () const
- void **setValue** (int value)
- int **getValueMax** () const

Public Attributes

- QString **styleTexte**

Protected Attributes

- int **value**
- int **valueMax**

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

- serveur/objet_virtuel.h
- serveur/objet_virtuel.cpp

52.23 qt_meta_stringdata_MainWindow_t Struct Reference

Public Attributes

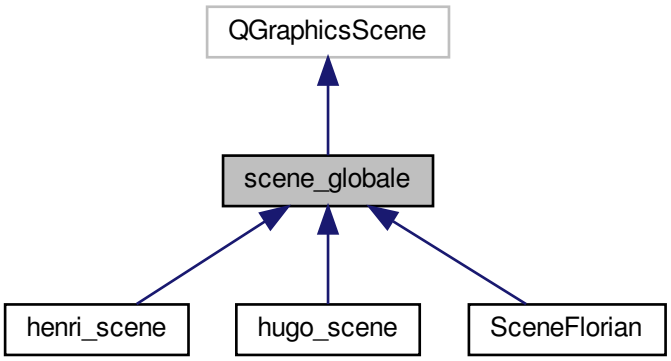
- QByteArrayData **data** [5]
- char **stringdata0** [42]

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

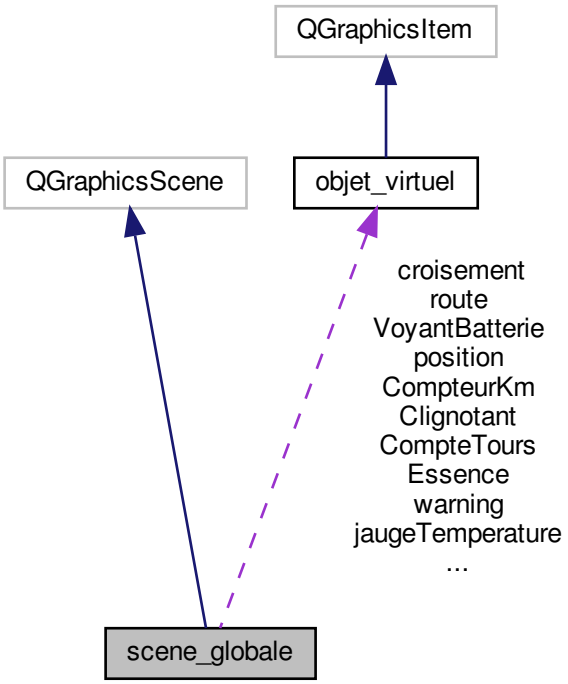
- build-serveur-Desktop-Debug/moc_mainwindow.cpp

52.24 scene_globale Class Reference

Inheritance diagram for scene_globale:



Collaboration diagram for scene_globale:



Public Member Functions

- **scene_globale** (QGraphicsScene ∗parent=nullptr)

Public Attributes

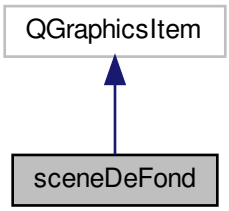
- **objet_virtuel** ∗ **Vitesse**
- **objet_virtuel** ∗ **Essence**
- **objet_virtuel** ∗ **CompteTours**
- **objet_virtuel** ∗ **jaugeTemperature**
- **objet_virtuel** ∗ **Clignotant**
- **objet_virtuel** ∗ **VoyantBatterie**
- **objet_virtuel** ∗ **position**
- **objet_virtuel** ∗ **croisement**
- **objet_virtuel** ∗ **route**
- **objet_virtuel** ∗ **warning**
- **objet_virtuel** ∗ **CompteurKm**

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

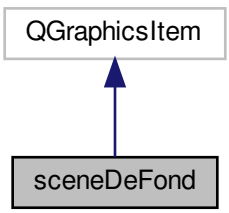
- serveur/scene_globale.h
- serveur/scene_globale.cpp

52.25 sceneDeFond Class Reference

Inheritance diagram for sceneDeFond:



Collaboration diagram for sceneDeFond:



Public Member Functions

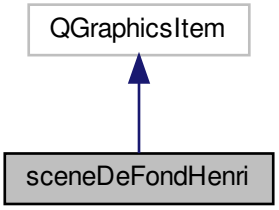
- **sceneDeFond** (QGraphicsItem ∗parent=nullptr)

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

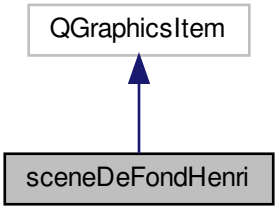
- serveur/Florian/scenedefond.h
- serveur/Florian/scenedefond.cpp

52.26 sceneDeFondHenri Class Reference

Inheritance diagram for sceneDeFondHenri:



Collaboration diagram for sceneDeFondHenri:



Public Member Functions

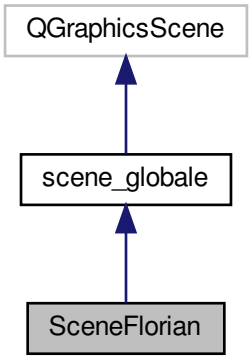
- **sceneDeFondHenri** (QGraphicsItem ∗parent=nullptr)
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)
- QRectF **boundingRect** () const

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

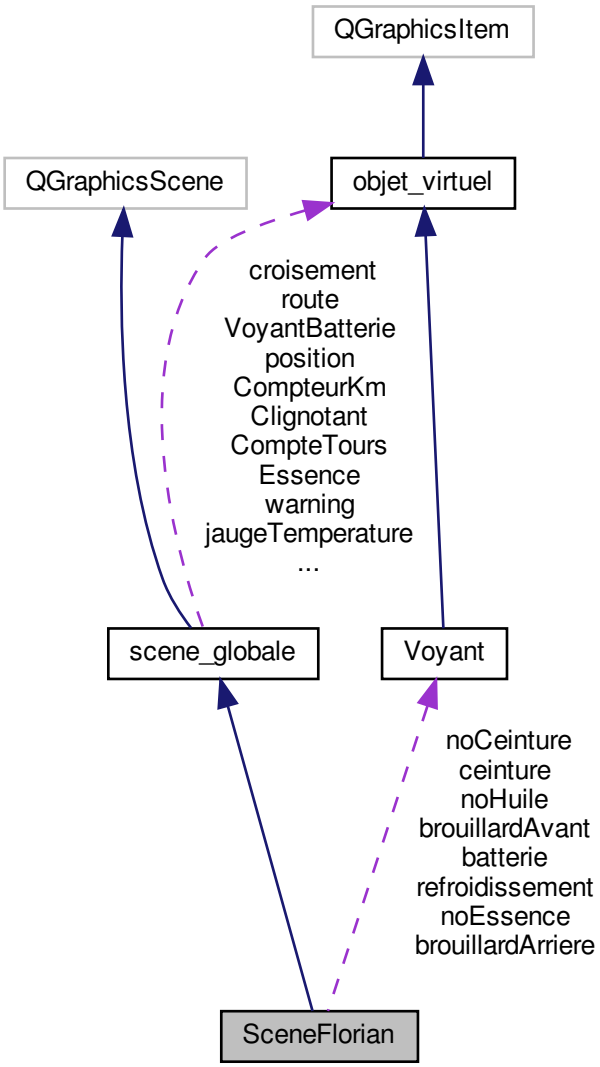
- serveur/Henri/scenedefondhenri.h
- serveur/Henri/scenedefondhenri.cpp

52.27 SceneFlorian Class Reference

Inheritance diagram for SceneFlorian:



Collaboration diagram for SceneFlorian:



Public Member Functions

- **SceneFlorian** (QGraphicsScene ∗parent=nullptr)

Public Attributes

- **Voyant ∗ brouillardAvant**
- **Voyant ∗ brouillardArriere**
- **Voyant ∗ noEssence**
- **Voyant ∗ noHuile**
- **Voyant ∗ refroidissement**

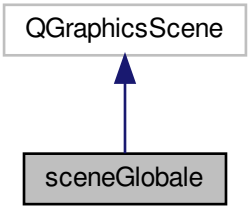
- **Voyant ∗ batterie**
- **Voyant ∗ ceinture**
- **Voyant ∗ noCeinture**

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

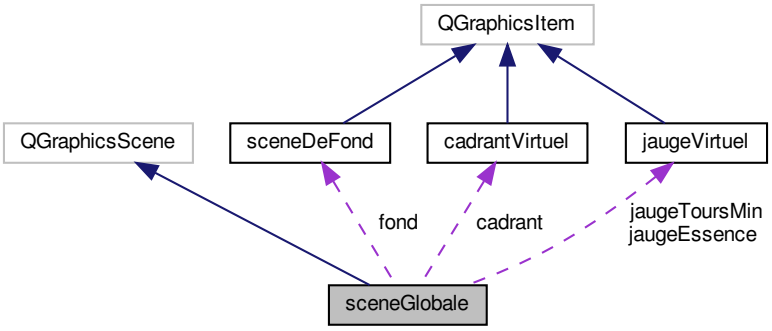
- serveur/Florian/sceneflorian.h
- serveur/Florian/sceneflorian.cpp

52.28 sceneGlobale Class Reference

Inheritance diagram for sceneGlobale:



Collaboration diagram for sceneGlobale:



Public Member Functions

- **sceneGlobale** (QGraphicsScene ∗parent=nullptr)

Public Attributes

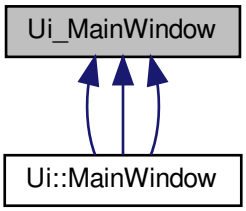
- **sceneDeFond** ∗ **fond**
- **cadrantVirtuel** ∗ **cadrant**
- **jaugeVirtuel** ∗ **jaugeEssence**
- **jaugeVirtuel** ∗ **jaugeToursMin**

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

- serveur/Florian/sceneglobale.h
- serveur/Florian/sceneglobale.cpp

52.29 Ui_MainWindow Class Reference

Inheritance diagram for Ui_MainWindow:



Public Member Functions

- void **setupUi** (QMainWindow ∗ **MainWindow**)
- void **retranslateUi** (QMainWindow ∗ **MainWindow**)
- void **setupUi** (QMainWindow ∗ **MainWindow**)
- void **retranslateUi** (QMainWindow ∗ **MainWindow**)
- void **setupUi** (QMainWindow ∗ **MainWindow**)
- void **retranslateUi** (QMainWindow ∗ **MainWindow**)

Public Attributes

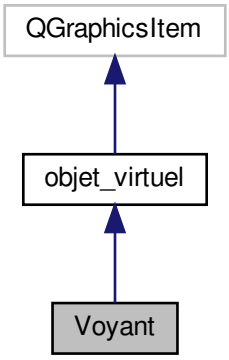
- QWidget ∗ **centralWidget**
- QVBoxLayout ∗ **verticalLayout**
- QGraphicsView ∗ **graphicsView**
- QMenuBar ∗ **menuBar**
- QToolBar ∗ **mainToolBar**
- QStatusBar ∗ **statusBar**

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

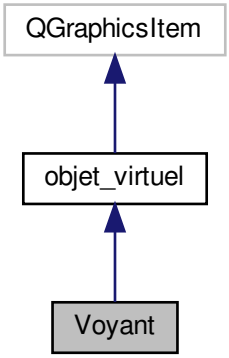
- build-serveur-Desktop-Debug/ui_mainwindow.h

52.30 Voyant Class Reference

Inheritance diagram for Voyant:



Collaboration diagram for Voyant:



Public Member Functions

- **Voyant** (QPixmap map, QGraphicsItem ∗parent=nullptr)
- QRectF **boundingRect** () const
- void **paint** (QPainter ∗painter, const QStyleOptionGraphicsItem ∗option, QWidget ∗widget)

Additional Inherited Members

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

- serveur/Florian/voyant.h
- serveur/Florian/voyant.cpp