

GPVehicle

De Wiki

Aller à : [navigation](#), [rechercher](#)
[GPVehicle](#)

Sommaire

- [1 How to call it](#)
 - [1.1 Dry mass initialization](#)
 - [1.2 Propulsive properties initialization](#)
 - [1.3 Shape characteristics initializaton](#)
 - [1.4 Aerodynamic properties initialization](#)
 - [1.5 Radiative properties initialization](#)
 - [1.6 Vehicle initialization](#)
- [2 Display](#)
- [3 How to use it](#)
- [4 How it is stored](#)

How to call it

For using the [GPVehicle](#) class, the developer has only to create such an object with these two possibilities:

- With no vehicle initialization

```
veh = new GPVehicle("My Vehicle");
```

- Or, if we want to initialize the widget with a predefined vehicle:

```
final CustomVehicle vehicle = new Vehicle(...);  
veh = new GPVehicle("My Vehicle", vehicle);
```

Since the **V2.0** version (and thus the [[PATRIUS](#)] V4.1.1), it is now possible to use directly the [Vehicle](#) class. So, it is possible to search in [[PATRIUS](#)] documentation how it is possible to initialize such an object. We have then to instantiate this class using all the previous sub-objects described in the following paragraphs.

*Note: for the previous **V1.3(.1)** versions, it was mandatory to use the specific **GENOPUS CustomVehicle** class.*

Dry mass initialization

```
// DRY MASS  
final double dryMass = 1000.;  
MassProperty dryMassProperty = new MassProperty(dryMass);
```

Propulsive properties initialization

```
// PROPULSION
```

```
final ArrayList<PropulsiveProperty> enginesList = new  
ArrayList<PropulsiveProperty>();  
PropulsiveProperty prop1 = new PropulsiveProperty(1000., 320.);  
prop1.setPartName("OCS");  
enginesList.add(prop1);  
PropulsiveProperty prop2 = new PropulsiveProperty(270., 150.);  
prop2.setPartName("ACS");  
enginesList.add(prop2);  
  
final ArrayList<TankProperty> fuelTankList = new ArrayList<TankProperty>();  
TankProperty tank = new TankProperty(500.);  
tank.setPartName("TANK");  
fuelTankList.add(tank);
```

*Note: for the previous **V1.3(.1)** versions, it was mandatory to use the specific **GENOPUS CustomEngine** and **CustomFuelTank** classes.*

Shape characteristics initializaton

```
// SHAPE (HERE PARALLELEPIPED + SOLAR PANELS)  
final RightParallelepiped parall = new RightParallelepiped(4.0, 1.0, 2.0);  
final RightParallelepiped solarPanels = new RightParallelepiped(10.0, 0.0,  
0.0);  
VehicleSurfaceModel vehicleRefSurface = new VehicleSurfaceModel(parall,  
solarPanels);
```

*Note: for the previous **V1.3(.1)** versions, it was mandatory to use the specific **GENOPUS** classes for shape (as for example **CustomParallelepiped** as well as the **CustomVehicleSurfaceModel** class."*

Aerodynamic properties initialization

```
// AERODYNAMIC PROPERTIES (CONSTANT COEFFICIENTS)  
final double cd = 2.0;  
final double cl = 0.;  
final CustomAerodynamicProperties aerodynamicProperties =  
    new CustomAerodynamicProperties(vehicleRefSurface, cd, cl);
```

Note #1: we need the shape model previously defined

*Note #2: for the previous **V1.3(.1)** versions, it was mandatory to use the specific **GENOPUS CustomAerodynamicProperties** class.*

Radiative properties initialization

```
// RADIATIVE PROPERTIES
final double ka = 1.0;
final double ks = 0.0;
final double kd = 0.0;
final RadiativeProperty rp = new RadiativeProperty(ka, ks, kd);

final double absorptionCoef = 1.0;
final double specularCoef = 0.0;
final double diffuseCoef = 0.0;
final RadiativeIRProperty rpIr =
    new RadiativeIRProperty(absorptionCoef, specularCoef, diffuseCoef);

final RadiativeProperties radiativeProperties =
    new RadiativeProperties(rp, rpIr, vehicleRefSurface);
```

Note #1: we need the shape model previously defined

*Note #2: for the previous **V1.3(.1)** versions, it was mandatory to use the specific **GENOPUS CustomRadiativeProperties** class.*

Vehicle initialization

At last, when all sub-objects are available, we have just to write this:

```
final Vehicle vehicle =
    new Vehicle(vehicleRefSurface, null, dryMassProperty,
                aerodynamicProperties, radiativeProperties,
                enginesList, fuelTankList);
```

Note: of course, possible to put null to some of them except for dry mass properties (a zero value for it will raise a warning as a negative one an error).

Display

With the previous initialization and by clicking on the engines and/or tanks for more information, we will have this display:

My Vehicle

Total mass: [kg](#)

Dry mass [kg](#)

Shape

Type: ☐ Sphere ☒ Parallelepiped ☐ Cylinder

Defined by: ☐ Dimension ☒ Surface

Length: [m](#)

Width: [m](#)

Height: [m](#)

X surface: [m²](#)

Y surface: [m²](#)

Z surface: [m²](#)

Solar panels ☐

Propulsive properties ☒

Ergol mass: [kg](#)

▶ ENGINES:

▶ TANKS:

Aerodynamic properties ☒

Drag Coefficient:

Lift Coefficient:

Radiative properties ☐

Propulsive properties
☒

Ergol mass:
kg

ENGINES:

Amount of engines

Engine number

[Items +/-](#)

Engine1

Engine

Name:

ISP:
s

Thrust:
N

TANKS:

Amount of fuel tanks

Tank number

[Items +/-](#)

Tank1

Fuel Tank

Name:

Propellant mass:
kg

How to use it

To get a [\[PATRIUS\]](#) [Vehicle](#) object, we will just have to call for the [getPatriusObject\(\)](#) method as below:

```
Vehicle vehicle = veh.getPatriusObject();
```

*Note: for the previous **V1.3(.1)** versions, the [getPatriusObject\(\)](#) method was returning a [CustomVehicle](#) object.*

How it is stored

Here is the [XML](#) format for such a vehicle:

```
<Vehicle name="My_Vehicle">
  <Boolean name="hasMassProperties">true</Boolean>
  <MassProperties name="massProperties">
```

```

    <Real name="dryMass" unit="kg">1.0E3</Real>
  </MassProperties>
  <Boolean name="hasPropulsiveProperties">true</Boolean>
  <PropulsiveProperties name="PropulsiveProperties">
    <Real name="ergMass" unit="kg">5.0E2</Real>
    <ComponentList name="listOfEngines">
      <!--Amount of items-->
      <Integer name="nbItems">2</Integer>
      <ComponentListItem name="Item_1">
        <Engine name="Engine">
          <String name="name">OCS</String>
          <Real name="isp" unit="s">3.2E2</Real>
          <Real name="thrust" unit="N">1.0E3</Real>
        </Engine>
      </ComponentListItem>
      <ComponentListItem name="Item_2">
        <Engine name="Engine">
          <String name="name">ACS</String>
          <Real name="isp" unit="s">1.5E2</Real>
          <Real name="thrust" unit="N">2.7E2</Real>
        </Engine>
      </ComponentListItem>
    </ComponentList>
    <ComponentList name="listOfTanks">
      <!--Amount of items-->
      <Integer name="nbItems">1</Integer>
      <ComponentListItem name="Item_1">
        <FuelTank name="Fuel_Tank">
          <String name="name">Tank1</String>
          <Real name="propMass" unit="kg">5.0E2</Real>
        </FuelTank>
      </ComponentListItem>
    </ComponentList>
  </PropulsiveProperties>
  <Boolean name="hasAerodynamicProperties">true</Boolean>
  <AerodynamicProperties name="AerodynamicProperties">
    <Real name="dragCoefficient">2.0E0</Real>
    <Real name="liftCoefficient">0.0E0</Real>
  </AerodynamicProperties>
  <Boolean name="hasRadiativeProperties">false</Boolean>
  <Shape name="VehicleShape">
    <String name="shapeType">Parallelepiped</String>
    <String name="shapeDefinitionType">Surface</String>
    <Real name="sx" unit="m^2">4.0E0</Real>
    <Real name="sy" unit="m^2">1.0E0</Real>
    <Real name="sz" unit="m^2">2.0E0</Real>
    <Boolean name="withSolarPanels">false</Boolean>
  </Shape>
</Vehicle>

```

Menu de navigation

Outils personnels

- [18.119.172.146](#)
- [Discussion avec cette adresse IP](#)
- [Créer un compte](#)
- [Se connecter](#)

Espaces de noms

- [Page](#)
- [Discussion](#)

Variantes

Affichages

- [Lire](#)
- [Voir le texte source](#)
- [Historique](#)
- [Exporter en PDF](#)

Plus

Rechercher

GENOPUS

- [Welcome](#)
- [Quick Start](#)
- [News](#)

User Manual

- [BasicPrinciples](#)
- [GPAbsoluteDate](#)
- [GPOrbit](#)
- [GPFramesConfiguration](#)

- [GPVehicle](#)
- [GPForceModels](#)
- [GPManeuverSequence](#)
- [GPAttitudeSequence](#)
- [GPIntegrator](#)
- [GPAxisCoordinates](#)
- [GPGeodeticPoint](#)
- [GPOneAxisEllipsoid](#)
- [GPRotation](#)
- [GPConstants](#)
- [Events](#)
- [GPCorrelation](#)

Evolutions

- [Main differences between V2.4.1 and V2.4.2](#)
- [Main differences between V2.3.3 and V2.4.1](#)
- [Main differences between V2.2.1 and V2.3.3](#)
- [Main differences between V2.2 and V2.2.1](#)
- [Main differences between V2.1.1 and V2.2](#)
- [Main differences between V2.1 and V2.1.1](#)
- [Main differences between V2.0.1 and V2.1](#)
- [Main differences between V2.0 and V2.0.1](#)
- [Main differences between V1.3.1 and V2.0](#)
- [Main differences between V1.3 and V1.3.1](#)
- [Main differences between V1.2.1 and V1.3](#)

Training

- [Make your own propagator tool!](#)
- [Tutorials package for V2.4.1](#)
- [Tutorials package for V2.3.3](#)
- [Tutorials package for V2.2](#)
- [Tutorials package for V2.1.1](#)
- [Tutorials package for V2.0 and V2.0.1](#)
- [Tutorials package for V1.3 and V1.3.1](#)
- [Training slides](#)

Links

- [CNES freeware server](#)

Outils

- [Pages liées](#)
- [Suivi des pages liées](#)
- [Pages spéciales](#)

- [Adresse de cette version](#)
- [Information sur la page](#)
- [Citer cette page](#)

- Dernière modification de cette page le 9 octobre 2018 à 15:38.

- [Politique de confidentialité](#)
- [À propos de Wiki](#)
- [Avertissements](#)

