

Dassault Mirage F1

BRÉMOND Lison, L2 Coursus Master en Ingénierie Physique, groupe A 2024-2025

lison.bremond@etu.sorbonne-universite.fr

Projet Modélisation 3D Mirage F1



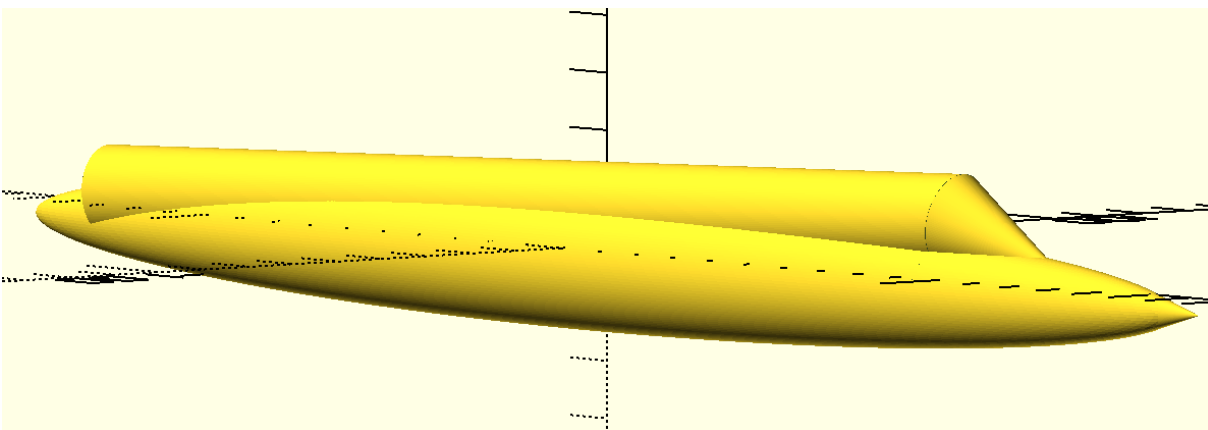
1. Fuselage

Pour poser les bases de la forme de notre avion, on vient dans un premier temps placer un cylindre "base", auquel on ajoute la pointe avant (appelée "nez" dans le code ci-dessous). On ajoute ensuite une longue tige qui rejoindra le cockpit à l'avant ("pare-brise" dans le code).

```
//fuselage base
module toutfuselage(){
  module fuselage(){
    translate([130,0,-2.5]) rotate([0,90,0]) scale([0.9,1,1]) cylinder(10,3,0.1,center=true);//nez
  };
  scale([1.2,0.8,1.2]) fuselage(); //allongement fuselage
  translate([0,0,-3]) rotate([0,0,0]) scale([13,0.8,1]) sphere(12,center=true);//pointe avant

  //cockpit
  module cockpit(){
    difference(){
      translate([-15,0,6]) rotate([0,88,0]) scale([0.8,0.65,1.1]) cylinder(220,10,10,center=true);//tige
      translate([110,0,6]) rotate([0,105,0]) scale([0.8,0.7,1]) cylinder(30,18,0.1,center=true);//pare brise
    };
    translate([110,0,6]) rotate([0,105,0]) scale([0.9,0.7,1]) cylinder(30,9.2,0.1,center=true);//pare brise
  };
  translate([0,0,0]) scale([1,1,1]) cockpit();
}
```

On obtient cette espèce de banane :



2. Entrées d'air

Ensuite, on ajoute les entrées d'air : des demi sphères allongées avec "scale", et trouées (avec une même forme de dimensions plus petites) à l'aide d'une fonction "difference".

```

//entrées d'air
module entreedair(){
module base(){
difference() {
translate([2,0,0]) scale([15,1,1.2]) sphere(10,center=true); //fuselage entrée d'air
translate([123,9,0]) scale([2,1.2,1.2]) cube(50,center=true); //cube de coupe largeur
translate([100,-14,0]) scale([10,0.5,1.2]) cube(50,center=true); //cube de coupe longueur
}};

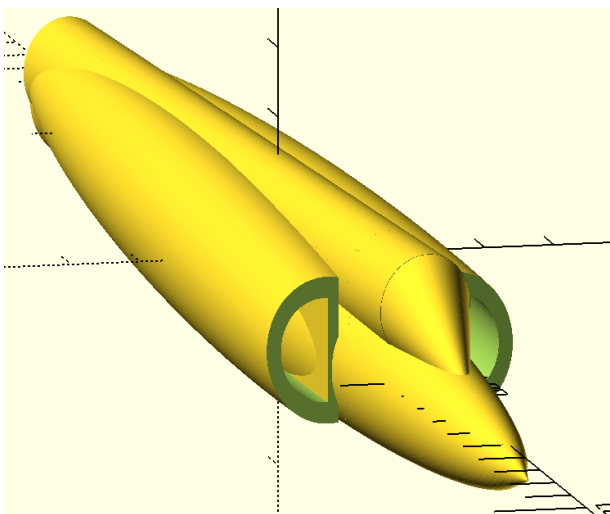
module creuxentreedair(){
difference() {
translate([2,0,0]) scale([15,1,1.2]) sphere(7.8,center=true); //fuselage entrée d'air
translate([123,9,0]) scale([2,1.2,1.2]) cube(50,center=true); //cube de coupe largeur
translate([100,-13,0]) scale([10,0.5,1.2]) cube(50,center=true); //cube de coupe longueur
}};

difference(){
rotate([0,0,2]) translate([0,5,0]) base();
rotate([0,0,2]) translate([1,5.5,0]) creuxentreedair();
}};

translate([0,1.5,0]) entreedair();
translate([0,-1.5,0]) rotate([180,0,0]) entreedair();

//réacteur
difference(){
translate([-130,0,-1.5]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(10,7,7,center = true);//réacteur
translate([-135,0,-1.5]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(5,6,6,center = true);//creux réacteur
};

```

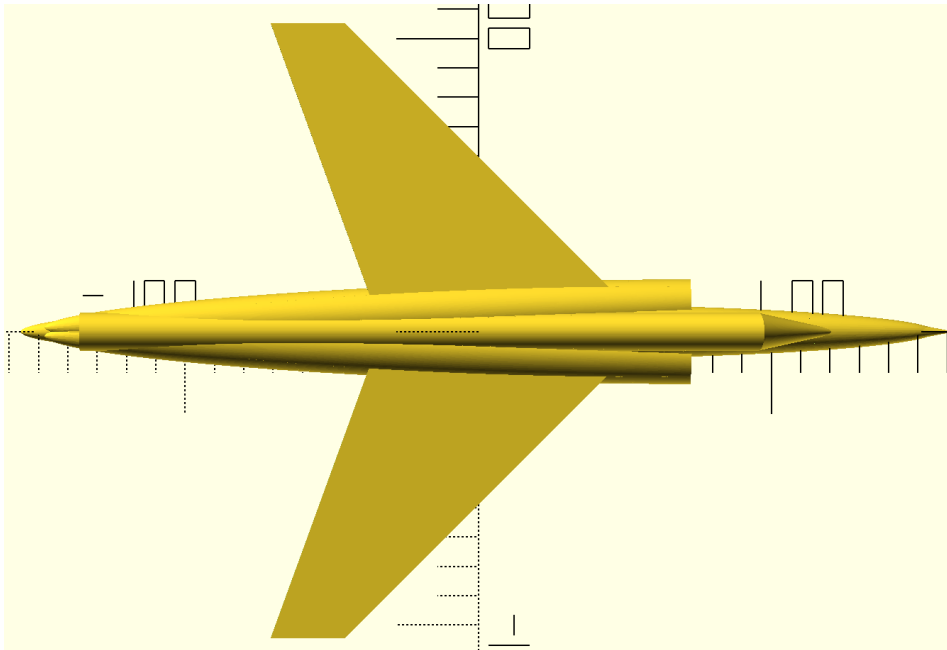


3. Grandes ailes, ailettes et aileron

Pour coder probablement la partie la plus importante de notre avion, on vient placer des cubes tous plats aux endroits des futures grandes ailes, que l'on vient sculpter avec d'autres cubes (déplacés avec "rotate" et "translate") grâce à la une longue fonction "difference".

```
//grandes ailes
module grandeaile(){
  difference(){
    translate([-15,56,10]) scale([1.5,1.2,0.04]) cube(100,center = true);//base plate
    translate([30,90,10]) rotate([0,0,45]) scale([1,2,0.2]) cube(100,center = true);//grande avant
    translate([-102,90,10]) rotate([0,0,20]) scale([0.5,2,0.2]) cube(100,center = true);//grande arrière
    translate([-60,130,10]) scale([0.5,0.5,0.2]) cube(100,center = true);//petite côté
  };
};
translate([10,0,-4]) rotate([-2,0,0]) grandeaile();
translate([10,0,16]) rotate([-178,0,0]) grandeaile();
```

Une nouvelle banane mais avec des ailes cette fois :



On procède de la même façon pour les ailettes et l'aileron, en les ajustant avec l'outil "rotate" :

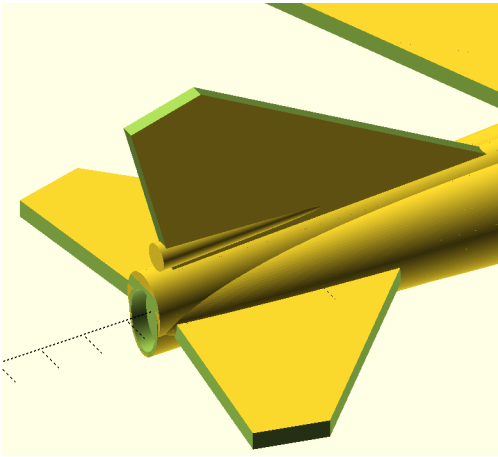
```
// ailette
module ailette(){
  difference(){
    translate([-15,56,10]) scale([1.5,1.2,0.04]) cube(100,center = true);//base plate
    translate([30,90,10]) rotate([0,0,45]) scale([1,2,0.2]) cube(100,center = true);//grande avant
    translate([-108,90,10]) rotate([0,0,8]) scale([0.5,2,0.2]) cube(100,center = true);//grande arrière
```

```

translate([-60,125,10]) rotate([0,0,14]) scale([0.7,0.5,0.2]) cube(100,center = true);//petite côté
});};
translate([-96,0,-12]) scale([0.5,0.5,1]) ailette();
rotate([180,0,0]) translate([-96,0,-8]) scale([0.5,0.5,1]) ailette();

//aileron
module aileron(){
difference(){
translate([-15,56,10]) scale([1.7,1.3,0.03]) cube(100,center = true);//base plate
translate([45,95,13]) rotate([0,0,60]) scale([1.2,2.2,0.2]) cube(100,center = true);//grande avant
translate([-127,90,10]) rotate([0,0,15]) scale([0.5,2,0.2]) cube(100,center = true);//grande arrière
translate([-100,110,15]) rotate([0,0,15]) scale([0.7,0.5,0.2]) cube(100,center = true);//petite côté
});};

```

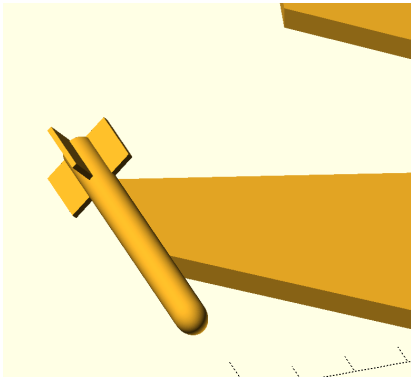


4. Missiles

Enfin, nous venons placer des missiles, en dessous et aux extrémités des grandes ailes. Pour cela, nous nous servirons de sphères allongées avec "scale" et de cylindres auxquels nous rajouterons des ailettes carrées. Le tout est de procéder symétriquement (selon l'axe y ici).

Latéraux

Pour les petits missiles latéraux, on se sert d'un cylindre et d'une sphère de même rayon à son extrémité. On ajoute à l'autre extrémité 2 plaquettes (cubes + fonction "scale" + "rotate" de 45° selon l'axe x).

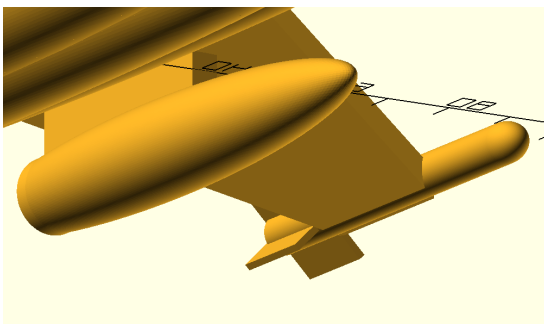


On crée un module "missileaile" pour pouvoir les déplacer facilement, puis on place nos cylindres en y et -y voulus avec 2 appels du module.

```
// missile extremité des ailes
module missileaile(){
  module missileailedep(){
    translate([0,0,50]) rotate([0,90,0]) cylinder(40,3,3,center=true); //base cylindre
    translate([20,0,50]) rotate([0,90,0]) sphere(3,center=true); //bout rond
    translate([-16,0,50]) rotate([45,0,0]) scale([0.9,1.5,0.1]) cube(10,center=true); //ailettes
    translate([-16,0,50]) rotate([-45,0,0]) scale([0.9,1.5,0.1]) cube(10,center=true); //ailettes
  };
  translate([0,0,-50]) missileailedep();};
  translate([-53,-103,3]) scale([1.5,1,1]) missileaile(); //aile gauche
  translate([-53,103,3]) scale([1.5,1,1]) missileaile(); //aile droite
```

NB : Le module "missileailedep" dans le code correspond à un premier module permettant de placer les missiles en [0,0,0] dans un premier temps (car ils n'ont pas été codés à l'origine du repère ; on ne verrait pas ce que l'on code sinon, puisqu'ils seraient "dans l'avion").

Inférieurs



Pour les plus gros missiles, on utilise un morceau de sphère allongée ("scale" toujours) que l'on viendra couper avec "difference". Sur le même principe que plus haut, on les place sous les

grandes ailes avec le même système de modules (nommés "grosboudin" dans le code). On place des maintiens entre les missiles et les grandes ailes à l'aide de parallélépipèdes.

```
// missile sous les ailes
module grosboudin(){
  module grosboudindep(){
    difference(){
      translate([0,0,50]) scale([5,0.7,0.7]) sphere(10,3,3,center=true); // base
      translate([0,0,50]) scale([5,0.7,0.7]) sphere(8,3,3,center=true); //vide
      translate([-45,0,50]) scale([4,3,3]) cube(8,center=true); // cube de coupe arrière
    };
  };
};
```

NB : - Pour alléger l'avion et réduire le temps d'impression, on "fait le vide" dans ces gros boudins avec une fonction "difference" et une sphère allongée plus petite placée aux mêmes coordonnées que la plus grande.

- On a coupé les grandes sphères (qui sont à présent vides) : il y a donc un trou béant à l'arrière de nos missiles. On vient alors placer un bouche-trous avec la fonction "circle" et ainsi avoir une grande sphère allongée plate sur une extrémité.

```
translate([-30,0,50]) rotate([0,90,0]) scale([0.7,0.7,3]) circle(8,center=true); // cercle de fermeture
};
translate([0,0,-50]) grosboudindep();
};
translate([-10,-40,-9]) grosboudin(); // gauche
translate([-10,40,-9]) grosboudin(); // droit
//maintiens
translate([-20,-40,0]) scale([5,0.5,1.5]) cube(8,center=true); // maintien gauche
translate([-20,40,0]) scale([5,0.5,1.5]) cube(8,center=true); // maintien droit
```

et voilà c tout !

5. Derniers détails

Pour le réacteur de l'avion, on vient couper l'arrière avec un cube et rajouter un cylindre troué. Cela permettra d'avoir une coupe propre en retirant les parties de fuselage et d'entrée d'air trop longues à la queue de l'avion.

```
// arrière
translate([-85,10,0]) rotate([90,0,0]) scale([0.55,0.55,1]) aileron();
translate([-133.5,0,-1]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(4,6,6,center = true); //comblage du trou
```

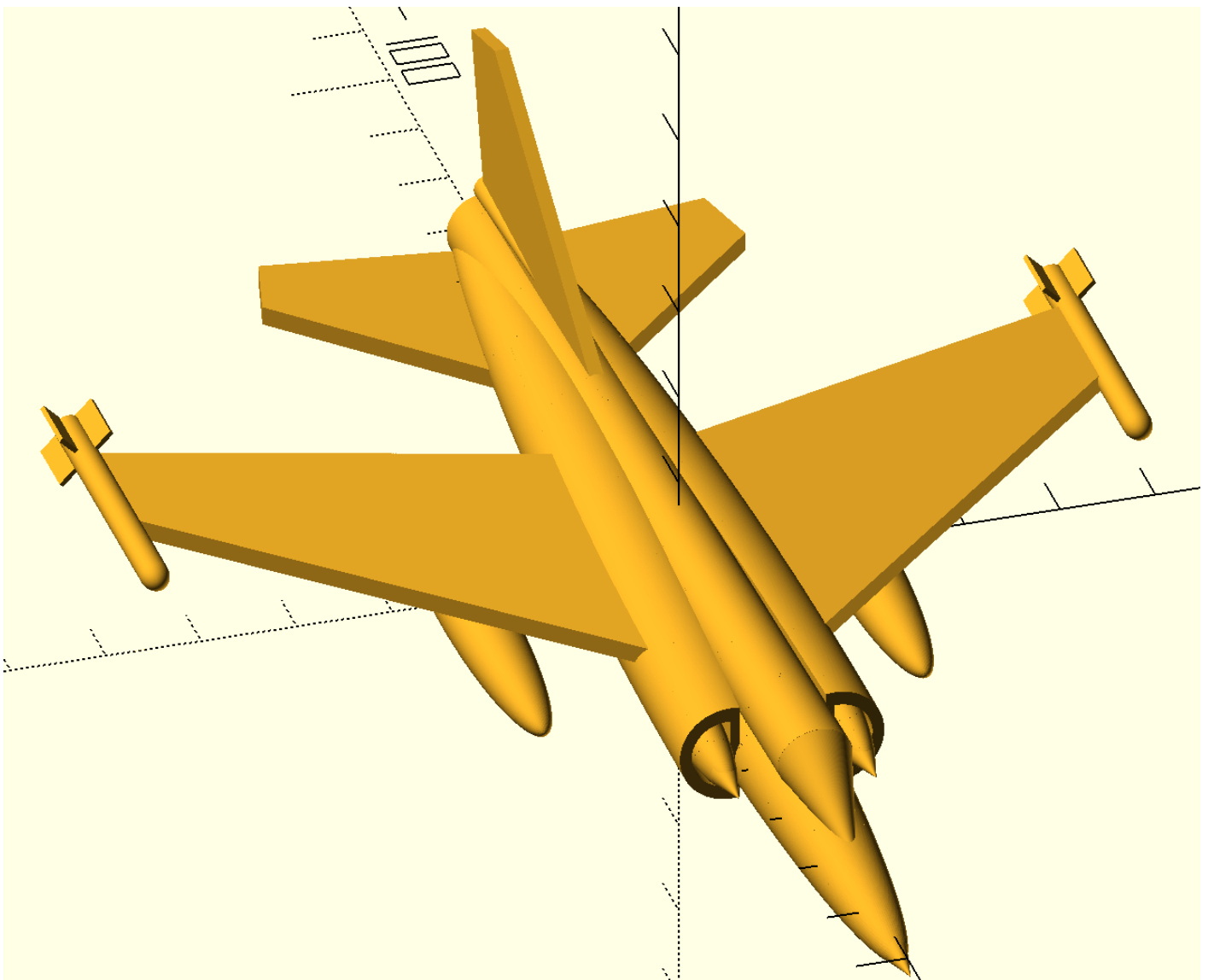
```

réacteur
translate([-108,0,13]) rotate([0,90,0]) scale([1,1,1]) cylinder(50,3,1,center=true);//détail aileron
};// fin module fuselage total

// coupe de l'arrière du fuselage
difference(){
toutfuselage();
translate([-156,0,-13]) scale([1,0.5,1]) cube(40,center = true); //cube de coupe
translate([-135,0,-1.5]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(4,6,6,center = true);//creux réacteur
};

```

6. Code complet



```

$fn=150;

module total(){

```



```

//fuselage base
module toutfuselage(){
  module elargissement(){
    module fuselage(){
      translate([130,0,-2.5]) rotate([0,90,0]) scale([0.9,1,1]) cylinder(10,3,0.1,center=true);//nez
    };
    scale([1.2,0.8,1.2]) fuselage(); //allongement fuselage
    translate([0,0,-3]) rotate([0,0,0]) scale([13,0.8,1]) sphere(12,center=true);//pointe avant

//cockpit
    module cockpit(){
      difference(){
        translate([-15,0,6]) rotate([0,88,0]) scale([0.8,0.65,1.1]) cylinder(220,10,10,center=true);//tige
        translate([110,0,6]) rotate([0,105,0]) scale([0.8,0.7,1]) cylinder(30,18,0.1,center=true);//pare brise
      };
      translate([110,0,6]) rotate([0,105,0]) scale([0.9,0.7,1]) cylinder(30,9.2,0.1,center=true);
    };
    translate([0,0,0]) scale([1,1,1]) cockpit();

    translate([55,12,0]) scale([3,0.4,0.6]) sphere(10,center=true); //cylindre entrée d'air droite
    translate([85,12,0]) rotate([0,90,0]) scale([1.3,1,1]) cylinder(10,2.3,0.1,center=true); //pointe ea droite
    translate([55,-12,0]) scale([3,0.4,0.6]) sphere(10,center=true); //cylindre entrée d'air gauche
    translate([85,-12,0]) rotate([0,90,0]) scale([1.3,1,1]) cylinder(10,2.3,0.1,center=true); //pointe ea gauche

//entrées d'air
    module entreedair(){
      module base(){
        difference() {
          translate([2,0,0]) scale([15,1,1.2]) sphere(10,center=true); //fuselage entrée d'air
          translate([123,9,0]) scale([2,1.2,1.2]) cube(50,center=true); //cube de coupe largeur
          translate([100,-14,0]) scale([10,0.5,1.2]) cube(50,center=true); //cube de coupe longueur
        };
      };

      module creuxentreedair(){
        difference() {
          translate([2,0,0]) scale([15,1,1.2]) sphere(7.8,center=true); //fuselage entrée d'air
          translate([123,9,0]) scale([2,1.2,1.2]) cube(50,center=true); //cube de coupe largeur
          translate([100,-13,0]) scale([10,0.5,1.2]) cube(50,center=true); //cube de coupe longueur
        };
      };
    };
  };
}

```

```

difference(){
rotate([0,0,2]) translate([0,5,0]) base();
rotate([0,0,2]) translate([1,5.5,0]) creuxentreedair();
};};
translate([0,1.5,0]) entreedair();
translate([0,-1.5,0]) rotate([180,0,0]) entreedair();

//réacteur
difference(){
translate([-130,0,-1.5]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(10,7,7,center = true);//réacteur
translate([-135,0,-1.5]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(5,6,6,center = true);//creux réacteur
};};
scale([1,1.2,1]) elargissement();

//grandes ailes
module grandeaile(){
difference(){
translate([-15,56,10]) scale([1.5,1.2,0.04]) cube(100,center = true);//base plate
translate([30,90,10]) rotate([0,0,45]) scale([1,2,0.2]) cube(100,center = true);//grande avant
translate([-102,90,10]) rotate([0,0,20]) scale([0.5,2,0.2]) cube(100,center = true);//grande arrière
translate([-60,130,10]) scale([0.5,0.5,0.2]) cube(100,center = true);//petite côté
};};
translate([10,0,-4]) rotate([-2,0,0]) grandeaile();
translate([10,0,16]) rotate([-178,0,0]) grandeaile();

// ailette
module ailette(){
difference(){
translate([-15,56,10]) scale([1.5,1.2,0.04]) cube(100,center = true);//base plate
translate([30,90,10]) rotate([0,0,45]) scale([1,2,0.2]) cube(100,center = true);//grande avant
translate([-108,90,10]) rotate([0,0,8]) scale([0.5,2,0.2]) cube(100,center = true);//grande arrière
translate([-60,125,10]) rotate([0,0,14]) scale([0.7,0.5,0.2]) cube(100,center = true);//petite côté
};};
translate([-96,0,-12]) scale([0.5,0.5,1]) ailette();
rotate([180,0,0]) translate([-96,0,-8]) scale([0.5,0.5,1]) ailette();

//aileron
module aileron(){
difference(){
translate([-15,56,10]) scale([1.7,1.3,0.03]) cube(100,center = true);//base plate

```

```

translate([45,95,13]) rotate([0,0,60]) scale([1.2,2.2,0.2]) cube(100,center = true); //grande avant
translate([-127,90,10]) rotate([0,0,15]) scale([0.5,2,0.2]) cube(100,center = true); //grande arrière
translate([-100,110,15]) rotate([0,0,15]) scale([0.7,0.5,0.2]) cube(100,center = true); //petite côté
};};

// arrière
translate([-85,10,0]) rotate([90,0,0]) scale([0.55,0.55,1]) aileron();
translate([-133.5,0,-1]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(4,6,6,center = true); //comblage du trou
réacteur
translate([-108,0,13]) rotate([0,90,0]) scale([1,1,1]) cylinder(50,3,1,center=true); //détail aileron
}; // fin module fuselage total

// coupe de l'arrière du fuselage
difference(){
toutfuselage();
translate([-156,0,-13]) scale([1,0.5,1]) cube(40,center = true); //cube de coupe
translate([-135,0,0]) rotate([0,90,0]) scale([1.2,0.8,1.2]) cylinder(4,6,6,center = true); //creux réacteur
};

// missile extrémité des ailes
module missileaile(){
module missileailedep(){
translate([0,0,50]) rotate([0,90,0]) cylinder(40,3,3,center=true); //base cylindre
translate([20,0,50]) rotate([0,90,0]) sphere(3,center=true); //bout rond
translate([-16,0,50]) rotate([45,0,0]) scale([0.9,1.5,0.1]) cube(10,center=true); //ailettes
translate([-16,0,50]) rotate([-45,0,0]) scale([0.9,1.5,0.1]) cube(10,center=true); //ailettes
};
translate([0,0,-50]) missileailedep();};
translate([-53,-103,3]) scale([1.5,1,1]) missileaile(); //aile gauche
translate([-53,103,3]) scale([1.5,1,1]) missileaile(); //aile droite

// missile sous les ailes
module grosboudin(){
module grosboudindep(){
difference(){
translate([0,0,50]) scale([5,0.7,0.7]) sphere(10,3,3,center=true); // base
translate([0,0,50]) scale([5,0.7,0.7]) sphere(8,3,3,center=true); //vide
translate([-45,0,50]) scale([4,3,3]) cube(8,center=true); // cube de coupe arrière
};

```

```
translate([-30,0,50]) rotate([0,90,0]) scale([0.7,0.7,3]) circle(8,center=true); // cercle de fermeture
};
translate([0,0,-50]) grosboudindep();
};
translate([-10,-40,-9]) grosboudin(); // gauche
translate([-10,40,-9]) grosboudin(); // droit

translate([-20,-40,0]) scale([5,0.5,1.5]) cube(8,center=true); // maintient gauche
translate([-20,40,0]) scale([5,0.5,1.5]) cube(8,center=true); // maintient droit
};
[scale([0.5,0.5,0.5]) total();
```



novembre 2024

Revision #3

Created 8 November 2024 20:32:57 by Lison

Updated 15 November 2024 20:09:16 by Lison