

```

public class ASLSash
{
    public bool IsUpper;
    public double TotZHeight, TotXWidth, YDepth, StileXZWidths, TopRailHeight, BottomRailHeight,
    GlazingBarWidths;
    public int PanesDown, PanesAcross;

    private SimpleSections SS = new SimpleSections();

    public ASLSash Clone()
    {
        return (ASLSash)this.MemberwiseClone();
        //OK as long as no independent pointers
    }

    public ASLMesh Draw()
    {
        //Draw Sash in XZ plane, centred on Z axis, bottom edge on X axis. Outside is plus Y, Inside minus
        Y.
        //Front inside face is Y = 0 plane

        ASLMesh M, M2, MOvolo;
        ASL3d pc, sz;
        ASL3dVector vx = new ASL3dVector(1, 0, 0);
        ASL3dVector vy = new ASL3dVector(0, 1, 0);
        ASL3dVector vz = new ASL3dVector(0, 0, 1);
        ASLBox B;
        double dd, spacing;
        int ii;

        //first outer frame, based on bottom rail
        pc = new ASL3d(0, YDepth/2, BottomRailHeight/2);
        sz = new ASL3d(TotXWidth, YDepth, BottomRailHeight);
        B = new ASLBox(pc, sz);
        M = B.Mesh();

        M.Material = new ASLMaterial();
        M.Material.Ambient = new ASLColour();
        M.Material.Ambient.R = M.Material.Ambient.G = 120;
        M.Material.Name = "WindowSash";
        M.SubMeshes = new ASLStack<ASLMesh>(3, 2);

        pc = new ASL3d(0, YDepth/2, TotZHeight - TopRailHeight/2);
        sz = new ASL3d(TotXWidth, YDepth, TopRailHeight);
        B = new ASLBox(pc, sz);
        M.SubMeshes.Add(B.Mesh());

        pc = new ASL3d((-TotXWidth + StileXZWidths)/2, YDepth/2, TotZHeight/2);
        sz = new ASL3d(StileXZWidths, YDepth, TotZHeight);
        B = new ASLBox(pc, sz);
        M2 = B.Mesh();
        M.SubMeshes.Add(M2);
        M2 = M2.Clone();
        M2.Translate((TotXWidth - StileXZWidths) * vx);
        M.SubMeshes.Add(M2);
    }
}

```

```

//next half-ovoloes lining frame
ASLPolyCurve CN = new ASLPolyCurve(); //(SS.HalfSashOvolo(), vx, false, false);
if (IsUpper)
{
    CN.IsClosed = false;
    CN.InitPoint = new ASL3dPoint(-TotXWidth / 2 + StileXZWidths, YDepth, BottomRailHeight);
//, GlazingBarWidths / 2, YDepth);
    CN.AddStraightLink(new ASL3dPoint(-TotXWidth / 2 + StileXZWidths, YDepth, TotZHeight -
TopRailHeight)); //, GlazingBarWidths / 2, YDepth);
    CN.AddStraightLink(new ASL3dPoint(TotXWidth / 2 - StileXZWidths, YDepth, TotZHeight -
TopRailHeight));
    CN.AddStraightLink( new ASL3dPoint(TotXWidth / 2 - StileXZWidths, YDepth,
BottomRailHeight)); //, GlazingBarWidths / 2, YDepth);
    //no Ovolo on Bottom Rail here
}
else
{
    CN.IsClosed = true;
    CN.InitPoint = new ASL3dPoint(-TotXWidth / 2 + StileXZWidths, YDepth, BottomRailHeight);
//, GlazingBarWidths / 2, YDepth);
    CN.AddStraightLink(new ASL3dPoint(-TotXWidth / 2 + StileXZWidths, YDepth, TotZHeight -
TopRailHeight)); //, GlazingBarWidths / 2, YDepth);
    CN.AddStraightLink(new ASL3dPoint(TotXWidth / 2 - StileXZWidths, YDepth, TotZHeight -
TopRailHeight)); //, GlazingBarWidths / 2, YDepth);
    CN.AddStraightLink(new ASL3dPoint(TotXWidth / 2 - StileXZWidths, YDepth,
BottomRailHeight));//, GlazingBarWidths / 2, YDepth);
}
    ASLExtrusion E = new ASLExtrusion(SS.HalfSashOvolo(), CN, 1, GlazingBarWidths / 2, YDepth,
CC.I, CC.J);
    //CN.Validate();
    M2 = E.Mesh();
    M.SubMeshes.Add(M2);

//and finally the Glazing Bars
if (PanessDown > 1)
{
    spacing = (TotZHeight - TopRailHeight - BottomRailHeight)/PanessDown;
    CN = new ASLPolyCurve(); //(SS.FullSashOvolo(), -1 * vz, false, false);
    CN.IsClosed = false;
    CN.InitPoint = new ASL3dPoint(-TotXWidth/2 + StileXZWidths, YDepth, 0);
    CN.AddStraightLink( new ASL3dPoint(TotXWidth/2 - StileXZWidths, YDepth, 0)); //,
GlazingBarWidths, YDepth);
    E = new ASLExtrusion(SS.FullSashOvolo(), CN, 1, GlazingBarWidths, YDepth, CC.I, CC.J);
    //CN.Validate();
    MOvolo = E.Mesh();
    for (ii = 1, dd = BottomRailHeight + spacing; ii < PanessDown; ii++, dd += spacing)
    {
        M2 = MOvolo.Clone();
        M2.Translate(dd * vz);
        M.SubMeshes.Add(M2);
    }
}
if (PanessAcross > 1)

```

```

{
    spacing = (TotXWidth - 2 * StileXZWidths)/PanelsAcross;
    CN = new ASLPolyCurve(); //SS.FullSashOvolo(), vx, false, false);
    CN.IsClosed = false;
    CN.InitPoint = new ASL3dPoint(0, YDepth, BottomRailHeight);
    CN.AddStraightLink( new ASL3dPoint(0, YDepth, TotZHeight - TopRailHeight)); //,
    GlazingBarWidths, YDepth);
    E = new ASLExtrusion(SS.FullSashOvolo(), CN, 1, GlazingBarWidths, YDepth, CC.I, CC.J);
    //CN.Validate();
    MOvolo = E.Mesh();
    for (ii = 1, dd = -TotXWidth/2 + StileXZWidths + spacing; ii < PanelsAcross; ii++, dd += spacing)
    {
        M2 = MOvolo.Clone();
        M2.Translate(dd * vx);
        M.SubMeshes.Add(M2);
    }
}

//and one sheet of glass
pc = new ASL3d(0, YDepth * 0.75, BottomRailHeight + (TotZHeight - BottomRailHeight -
TopRailHeight) / 2);
//the Y positioning here is a bit of a guesstimate, reckoning the rebate is about 1/3rd of YDepth,
so the inside face
//should be at YDepth/2 - YDepth/3 = YDepth/6. If the glass is YDepth/6 thick, YDepth/4 -
1/2*YDepth/6
//= (3/12 - 1/12) * YDepth = YDepth/6. OK.
sz = new ASL3d(TotXWidth - 2 * StileXZWidths, YDepth / 6, TotZHeight - BottomRailHeight -
TopRailHeight);
B = new ASLBox(pc, sz);
M2 = B.Mesh();

M2.Material = new ASLMaterial();
M2.Material.Ambient = new ASLColour();
M2.Material.Ambient.B = 200; M.Material.Ambient.G = 120;
M2.Material.Name = "WindowSashGlass";

M.SubMeshes.Add(M2);

return M;
}
}

```