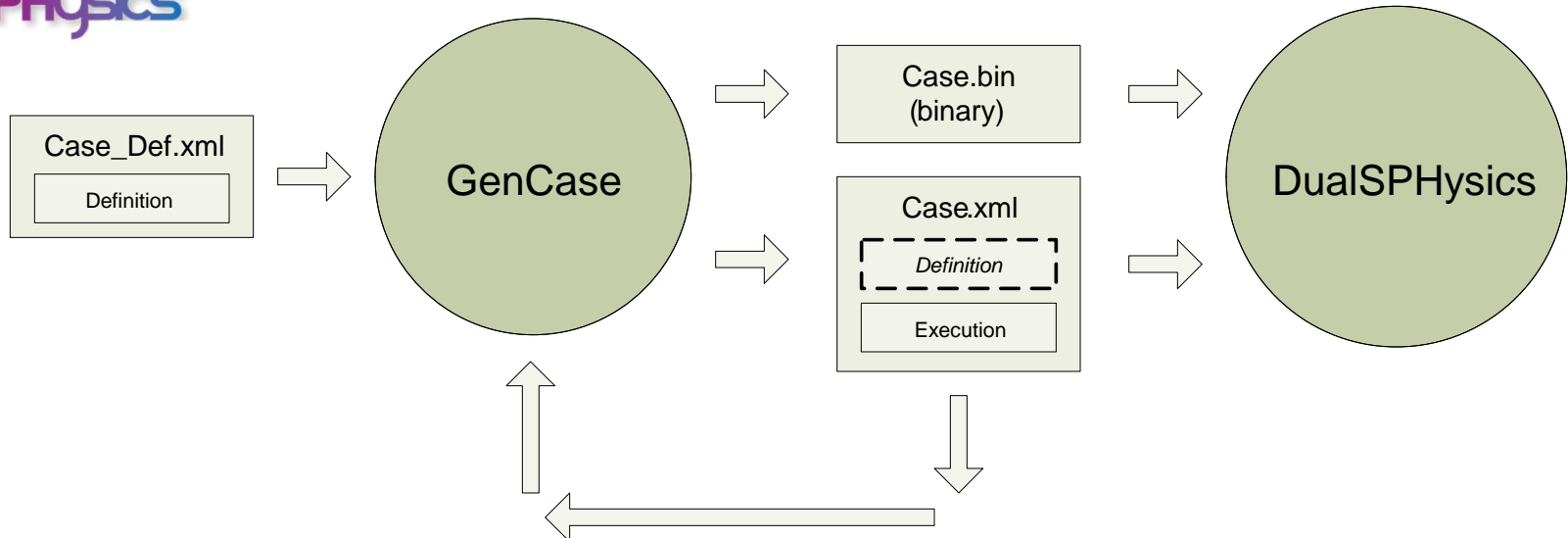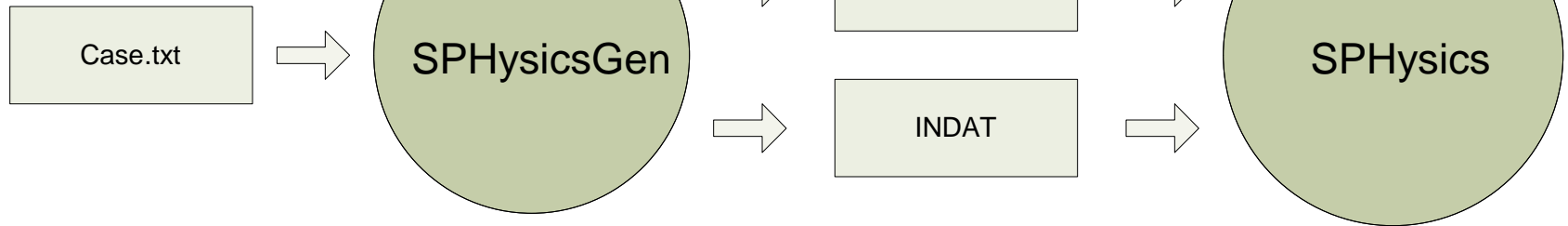# GENCASE XML GUIDE

Help to create your own case using the XML file.

December 2013

A.J.C. Crespo (alexbexe@uvigo.es)
J.M. Domínguez (jmdominguez@uvigo.es)
M.G. Gesteira (mggesteira@uvigo.es)
A. Barreiro (anxo.barreiro@uvigo.es)
B.D. Rogers (benedict.rogers@manchester.ac.uk)

# INPUT & OUTPUT FILES FOR SPHYSICSGEN AND GENCASE

# STRUCTURE OF THE XML FILE

It is divided in two sections:

- **"casedef"**  Definition of the case with the initial geometry and configuration.
  Created and used by GenCase

- **"execution"**  Information required to execute the case.
  Only to be used by DualSPHysics

```
- <case app="GenCase v0.88 (26-01-2011)" date="26-01-2011 19:31:40">
  - <casedef>
    + <constantsdef></constantsdef>
    + <mkconfig boundcount="240" fluidcount="10"></mkconfig>
    - <geometry>
      + <definition dp="0.05"></definition>
      - <commands>
        + <list name="Hello"></list>
        + <mainlist></mainlist>
        </commands>
      </geometry>
    + <initials></initials>
    + <floatings></floatings>
    + <motion></motion>
    </casedef>
  - <execution>
    + <parameters></parameters>
    </execution>
  </case>
```

# STRUCTURE OF THE XML FILE

- **"casedef"** :

  - **constantsdef**  constants needed in SPH
  - **mkconfig**      configuration of labels
  - **geometry**      geometry of the system (boundaries and fluid)
    - **definition**
    - **commands (list & mainlist)**
  - **initials**      special features for fluid particles
  - **floatings**     description of floating objects
  - **motion**        description of the movement of boundaries

- **"execution"**

  - **parameters**    parameters of execution in DualSPHysics

```
- <casedef>
    + <constantsdef></constantsdef>
    + <mkconfig boundcount="240" fluidcount="10"></mkconfig>
    - <geometry>
      + <definition dp="0.05"></definition>
      - <commands>
        + <list name="Hello"></list>
        + <mainlist></mainlist>
      </commands>
    </geometry>
    + <initials></initials>
    + <floatings></floatings>
    + <motion></motion>
  </casedef>
- <execution>
    + <parameters></parameters>
  </execution>
</case>
```

# STRUCTURE OF THE XML FILE

**CASEDEF**-**CONSTANTSDEF**

**CASEDEF**-**MKCONFIG**

**CASEDEF**-**GEOMETRY**-**DEFINITION**

**CASEDEF**-**GEOMETRY**-**COMMANDS**-**MAINLIST**

**CASEDEF**-**INITIALS**

**CASEDEF**-**FLOATINGS**

**CASEDEF**-**PROPERTIES**

**CASEDEF**-**MOTION**

**EXECUTION**-**PARAMETERS**

# CASEDEF-CONSTANTSDEF

```
- <constantsdef>
    <lattice bound="2" fluid="1"/>
    <gravity x="0" y="0" z="-9.81"/>
    <cflnumber value="0.2"/>
    <hswl value="0" auto="true"/>
    <coefsound value="10"/>
    <coefficient value="0.866025"/>
    <gamma value="7"/>
    <rhop0 value="1000"/>
    <eps value="0.5"/>
</constantsdef>
```

**lattice:** indicates the type of mesh to create particles
- 1: one particle per point
- 2: two particles per point
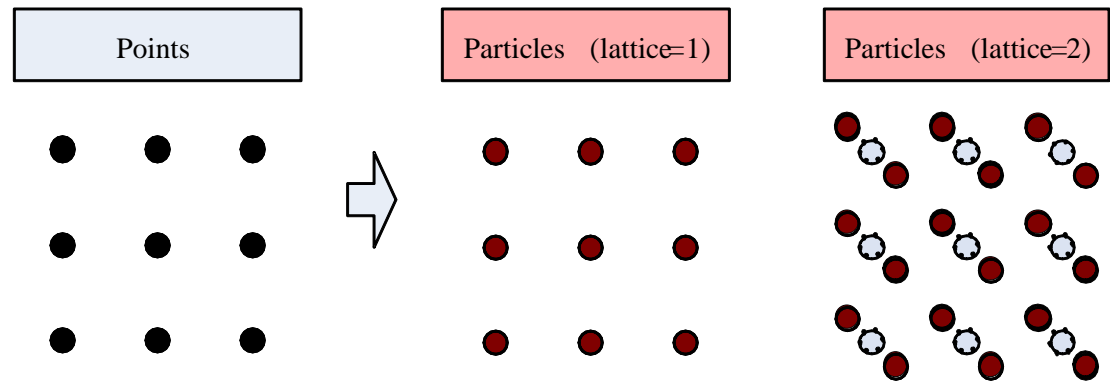


**gravity:** gravity acceleration

**cflnumber:** involved in the computation of the variable time step (0.1-0.3)

**hswl:** maximum water height in the system (automatically computed if *true*)

**coefsound:** $c_0 = coefsound * V_{MAX}$   (10-25)

**coefficient:** $h = coefficient * raiz(dx^2 + dy^2 + dz^2)$     (for 2D case *dy=0*)

**gamma:** 7

# CASEDEF-MKCONFIG

```
- <mkconfig boundcount="240" fluidcount="10">
    <mkorientbound mk="0" orient="YxZ"/>
    <mkorientfluid mk="1" orient="yzX"/>
    <mkorientfluid mk="2" orient="ZYx"/>
  </mkconfig>
```
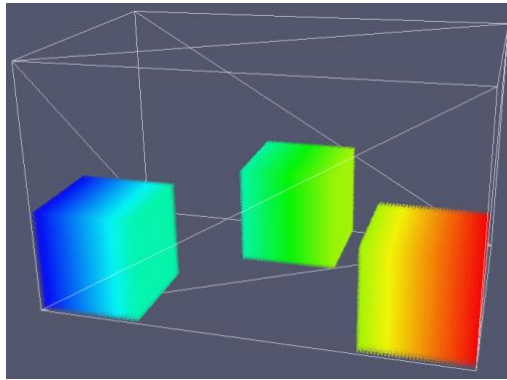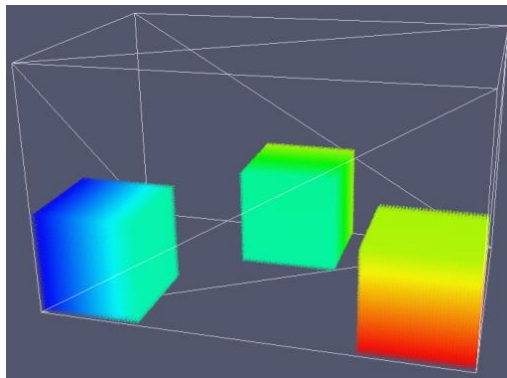
**mk:** label used to

- define the order of creation of the objects

- apply specific features to the different set of points such as movement, rigid motion…

240 labels for boundary particles and

10 labes for fluid particles

mkorientfluid = "xyz"



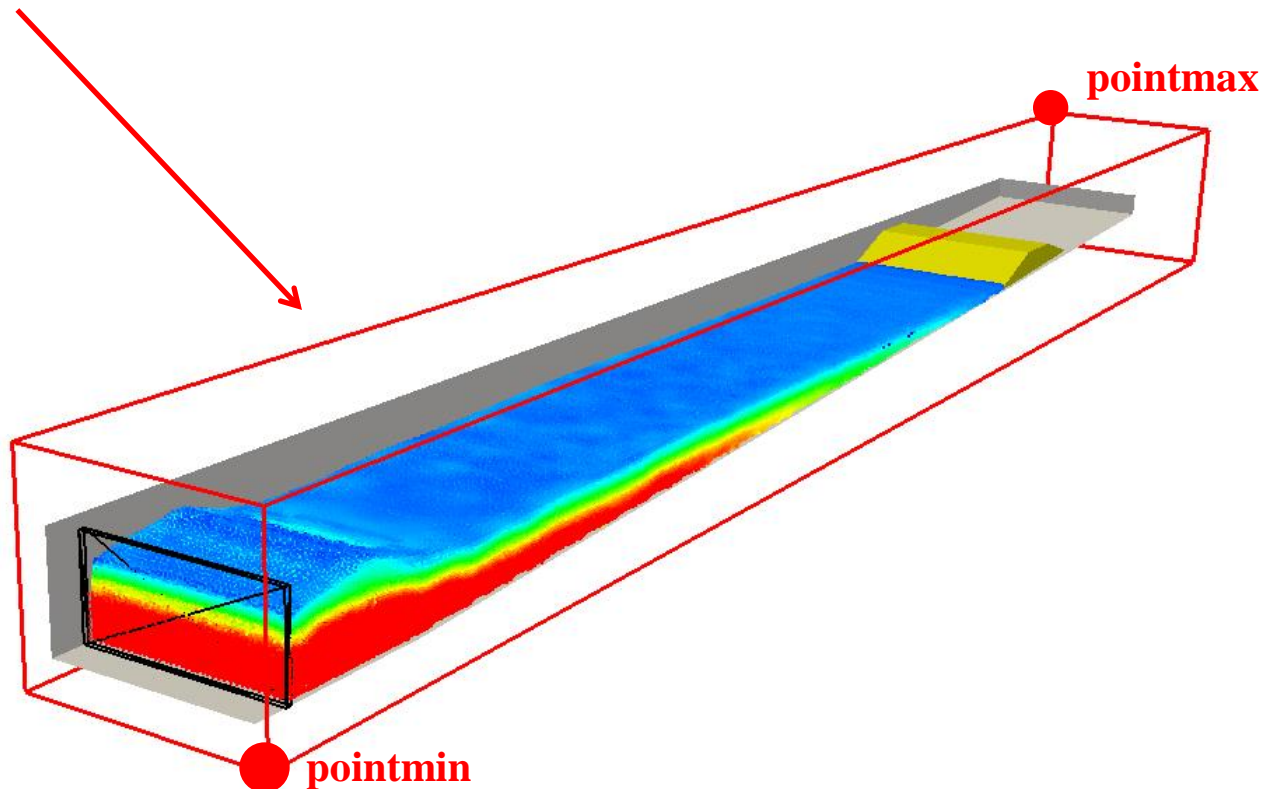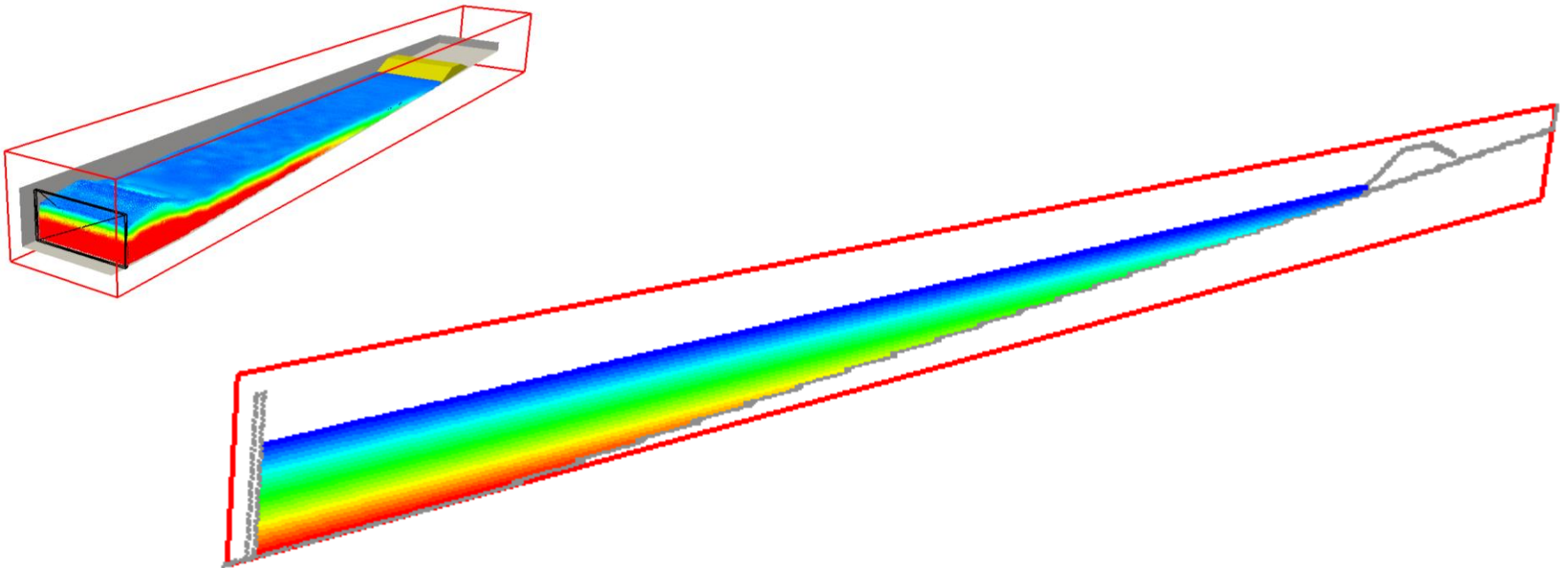mkorientfluid = "xyz"
mkorientfluid = "yzX"
mkorientfluid = "ZYx"



**mkorientation:** determines the order of particles when creating one object (useful for visualization with the variable *id*)

# CASEDEF-GEOMETRY-DEFINITION

```
− <definition dp="0.05">
    <pointmin x="-0.05" y="-0.05" z="-0.05"/>
    <pointmax x="2" y="1" z="1"/>
  </definition>
```

**dp** defines the distance between particles

**pointmin & pointmax** defines the dimensions of the domain where particles can be created

# CASEDEF-GEOMETRY-DEFINITION

```
− <definition dp="0.05">
    <pointmin x="-0.05" y="1" z="-0.05"/>
    <pointmax x="2" y="1" z="1"/>
  </definition>
```
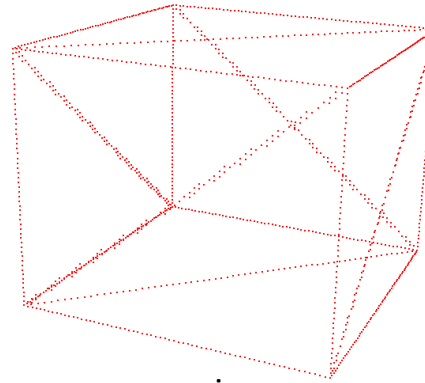
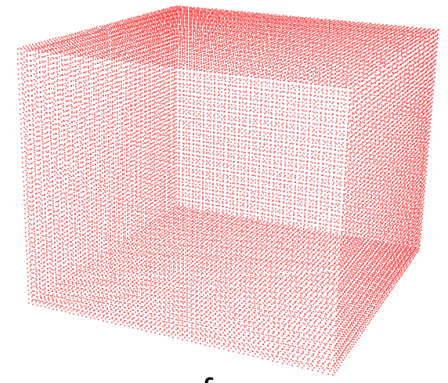A 2D configuration can be generated by imposing the same values along Y-direction

**\<pointmin\> = \<pointmax\>**

# CASEDEF-GEOMETRY-COMMANDS-MAINLIST
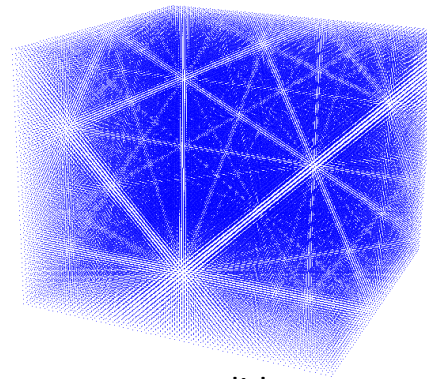
```xml
– <mainlist>
    <setdrawmode mode="wire"/>
    <setdrawmode mode="face"/>
    <setdrawmode mode="solid"/>
    <setdrawmode mode="full"/>
  </mainlist>
```
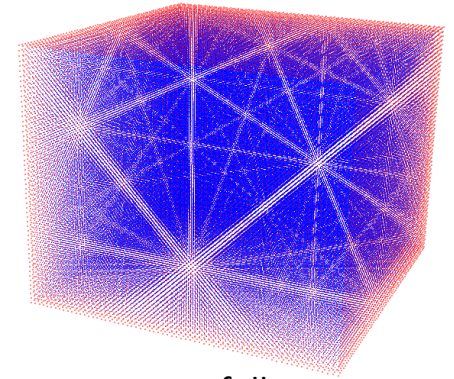


wire



face

**&lt;setdrawmode&gt;**:

  – "**wire**": wire mode
  – "**face**": draw faces
  – "**solid**": draw inside
  – "**full**": combines *face* and *solid*



solid



full

10

```
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setshapemode>dp | bound | fluid</setshapemode>
    <setshapemode>real | void</setshapemode>
  </mainlist>
```

**<setshapemode>**: defines the draw operations to create a VTK files (polygons)

- – "**real**": using the real coordinates
- – "**dp**": adjusting coordinates to *dp*
- – "**fluid**": operations with *mk-fluid.*
- – "**bound**": operations with *mk-bound*.
- – "**void**": operations with *mk-void*.

**\<setshapemode\>**: defines the draw operations to create a VTK files (polygons)

```
-<mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setdrawmode mode="full"/>
    <setmkfluid mk="0"/>
  +<drawbox></drawbox>
    <setmkbound mk="0"/>
  +<drawbox></drawbox>
    <shapeout file="Box"/>
    <setmkvoid/>
  +<drawbox></drawbox>
    <setmkbound mk="1"/>
  +<drawbox></drawbox>
    <shapeout file="Building"/>
</mainlist>
```

```
-<mainlist>
    <setshapemode>real | dp | bound</setshapemode>
    <setdrawmode mode="full"/>
    <setmkfluid mk="0"/>
  +<drawprism mask="0"></drawprism>
    <setmkvoid/>
  +<drawbox></drawbox>
    <setdrawmode mode="face"/>
    <setmkbound mk="10"/>
  +<drawbox></drawbox>
    <setmkbound mk="0"/>
  +<drawprism mask="96"></drawprism>
    <shapeout file="" reset="true"/>
</mainlist>
```

**shapeout**: creates VTK files (polygons)
of only some *bound* objects
Case_Box_Dp.vtk
Case_Building_Dp.vtk

**shapeout**: creates VTK files (polygons)
of all the *bound* objects
Case__Real.vtk
Case__Dp.vtk

**reset="true"** indicates that you will create a new VTK with **shapeout**
including objects created starting from this line

```
- <commands>
    - <mainlist>
        <setmkvoid/>
        <setmkfluid mk="0"/>
        <setmkbound mk="0"/>
        <setmknextfluid next="true"/>
        <setmknextbound next="false"/>
        <setmknextauto active="true"/>
    </mainlist>
</commands>
```

**\<setmkvoid\>**, **\<setmkfluid\>**, **\<setmkbound\>**: defines the label *mk* to draw points of type:

void (empty), fluid, bound

**\<setmknextfluid\>**, **\<setmknextbound\>**: increases (decreases) the value of *mk* with next=*true* (=*false*)

**\<setmknextauto\>**: after each draw command *mk* is increased automatically

# CASEDEF-GEOMETRY-COMMANDS-MAINLIST

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
    <move x="0.5" y="0" z="0"/>
  + <drawbox></drawbox>
    <shapeout file="BoxMove" reset="true"/>
    <matrixreset/>
    <scale x="2" y="1.5" z="0.5"/>
  + <drawbox></drawbox>
    <shapeout file="BoxScale" reset="true"/>
    <matrixreset/>
    <rotate x="0" y="0" z="1" ang="45"/>
  + <drawbox></drawbox>
    <shapeout file="BoxRotate" reset="true"/>
  </mainlist>
```
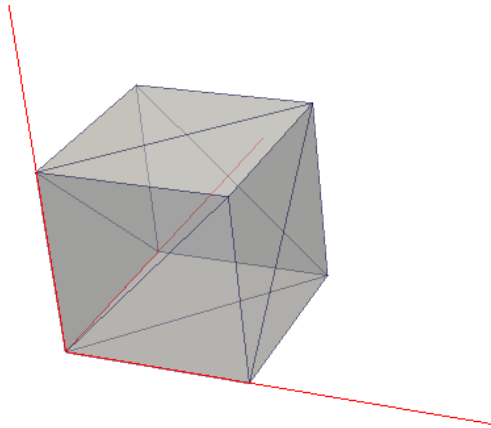
**<move>**: a displacement is applied to the transformation matrix

**<scale>**: scaling is applied to matrix

**<rotate>**: a rotation is given starting from a vector and an angle

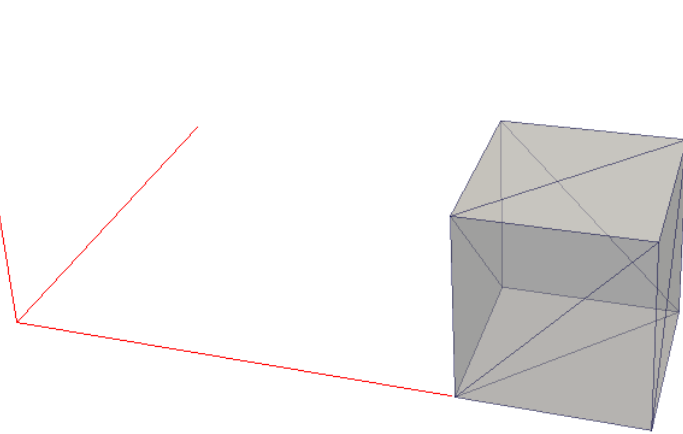**<matrixreset>**: the modified matrix is replace by the original one (identity matrix)

<drawbox …>

<move x="0.5" y="0" z="0"/>
 <drawbox …>

<scale x="2" y="1.5" z="0.5"/>
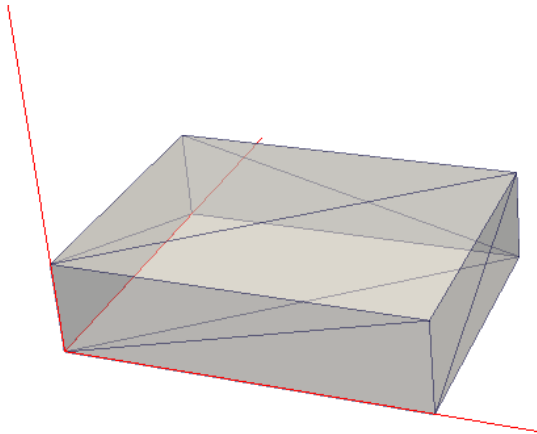<drawbox …>

<rotate x="0" y="0" z="1" ang="45"/>
<drawbox …>

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <setlinebegin>
      <point x="0" y="0" z="0"/>
    </setlinebegin>
  - <drawlineto>
      <point x="0" y="1" z="0"/>
    </drawlineto>
    <setmknextbound next="true"/>
  - <drawline>
      <point x="0" y="1" z="0"/>
      <point x="1" y="1" z="0"/>
    </drawline>
    <setmknextbound next="true"/>
  - <drawline>
      <point x="1" y="1" z="0"/>
      <point x="1" y="0" z="0"/>
    </drawline>
    <setmknextbound next="true"/>
  - <drawlines>
      <point x="1" y="0" z="0"/>
      <point x="0" y="0" z="0.5"/>
      <point x="0" y="1" z="0.5"/>
      <point x="1" y="1" z="0.5"/>
      <point x="1" y="0" z="0.5"/>
    </drawlines>
    <shapeout file="Lines" reset="true"/>
</mainlist>
```
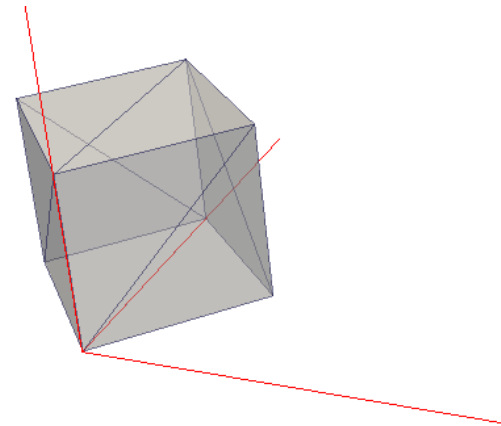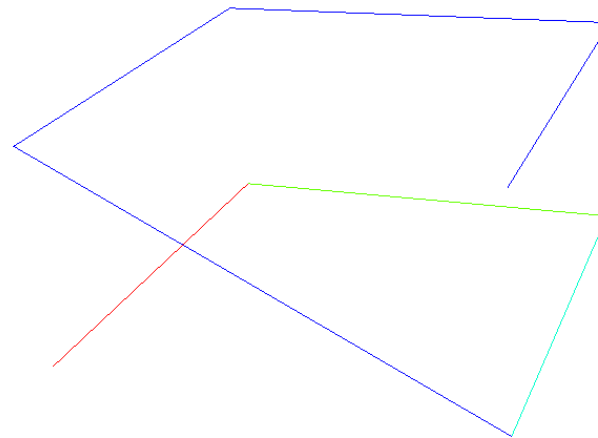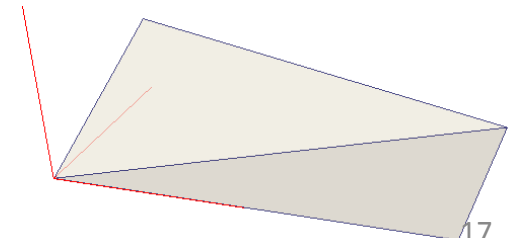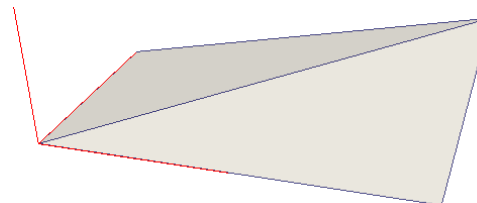
**<setlinebegin>**: sets the begining of the line with <drawlineto>

**<drawlineto>**: draws a line to a given point

**<drawline>**: draws a line between two points

**<drawlines>**: draws lines between several points

16

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <drawtriangle>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0" y="0.5" z="0"/>
    </drawtriangle>
    <shapeout file="Triangle" reset="true"/>
  - <drawquadri>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="1" y="0.5" z="0"/>
      <point x="0" y="0.5" z="0"/>
    </drawquadri>
    <shapeout file="Quadri" reset="true"/>
  - <drawquadri>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="1" y="0.5" z="0.2"/>
      <point x="0" y="0.5" z="0"/>
    </drawquadri>
    <shapeout file="Quadri2" reset="true"/>
  - <drawquadri>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="1" y="0.5" z="0"/>
      <point x="0" y="0.5" z="0.2"/>
    </drawquadri>
    <shapeout file="Quadri3" reset="true"/>
</mainlist>
```
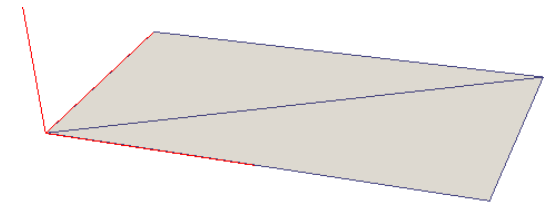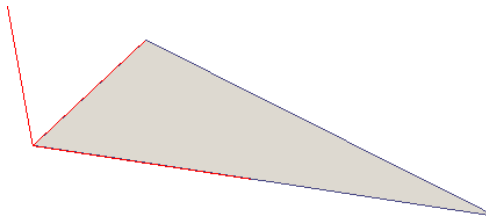
**<drawtriangle>**:  draws a triangle with tree points (points must always go counterclockwise)

**<drawquadri>**: draws the quadrilateral described by four points (points may not be in the same plane)



17

```
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <drawtrianglesstrip>
        <point x="0" y="1" z="0"/>
        <point x="0" y="0" z="0"/>
        <point x="1" y="1" z="0"/>
        <point x="1" y="0" z="0"/>
        <point x="2" y="1" z="0"/>
        <point x="2" y="0" z="0"/>
        <point x="3" y="1" z="0"/>
        <point x="3" y="0" z="0"/>
        <point x="4" y="1" z="0"/>
    </drawtrianglesstrip>
    <shapeout file="TrianglesStrip9" reset="true"/>
  </mainlist>
```

**<drawtrianglesstrip>**: draws a series of chained triangles

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <drawtrianglesfan>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0.9" y="0.5" z="0"/>
      <point x="0.5" y="0.9" z="0"/>
      <point x="0" y="1" z="0"/>
      <point x="-0.5" y="0.9" z="0"/>
    </drawtrianglesfan>
    <shapeout file="TrianglesFan" reset="true"/>
    <setmkbound mk="0"/>
  - <drawtrianglesfan>
      <point x="0" y="0" z="1"/>
      <point x="1" y="0" z="0"/>
      <point x="0.8" y="0.6" z="0"/>
      <point x="0.2" y="1" z="0"/>
      <point x="-0.5" y="0.9" z="0"/>
      <point x="-0.9" y="0.3" z="0"/>
      <point x="-0.9" y="-0.3" z="0"/>
      <point x="-0.5" y="-0.9" z="0"/>
      <point x="0.2" y="-1" z="0"/>
      <point x="0.8" y="-0.6" z="0"/>
      <point x="1" y="0" z="0"/>
    </drawtrianglesfan>
    <shapeout file="TrianglesFan2" reset="true"/>
</mainlist>
```
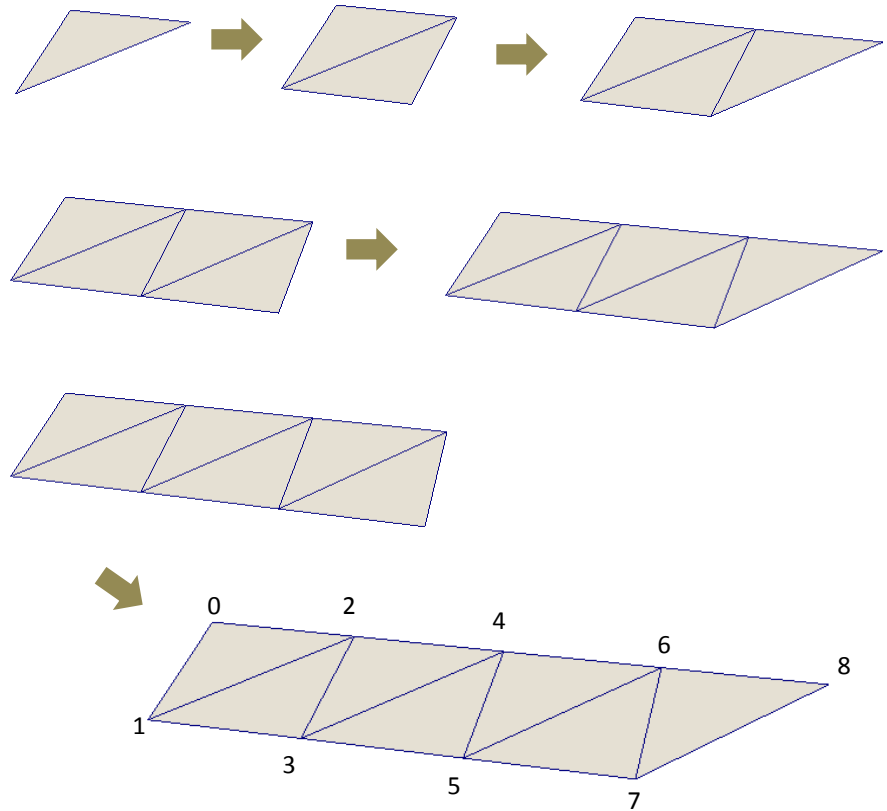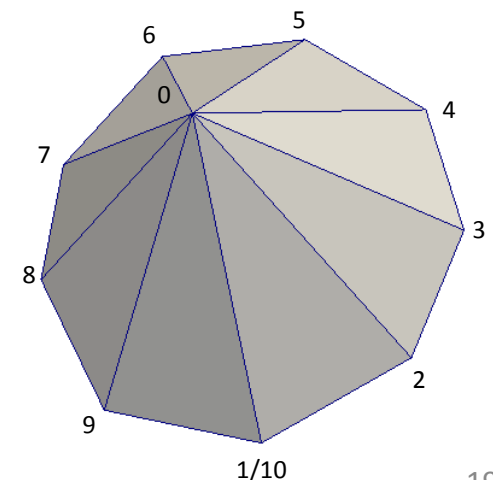
**<drawtrianglesfan>**: draws a range of triangles



19

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <drawtriangles>
    - <points>
        <point x="0" y="0" z="0"/>
        <point x="1" y="0" z="0"/>
        <point x="1" y="1" z="0"/>
        <point x="0" y="1" z="0"/>
        <point x="0" y="0" z="0.8"/>
        <point x="1" y="0" z="0.8"/>
        <point x="1" y="1" z="0.8"/>
        <point x="0" y="1" z="0.8"/>
      </points>
    - <triangles>
        <triangle x="0" y="1" z="5"/>
        <triangle x="1" y="2" z="6"/>
        <triangle x="2" y="3" z="7"/>
        <triangle x="3" y="0" z="4"/>
        <triangle x="0" y="2" z="1"/>
        <triangle x="4" y="5" z="6"/>
      </triangles>
    </drawtriangles>
    <shapeout file="Triangles" reset="true"/>
</mainlist>
```
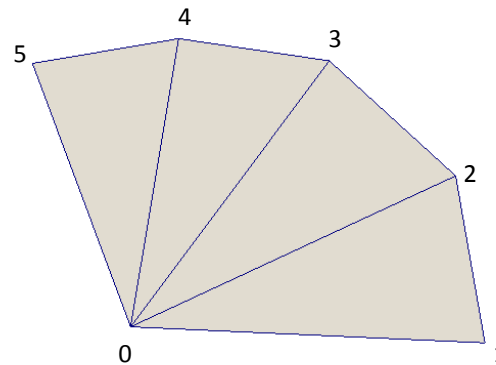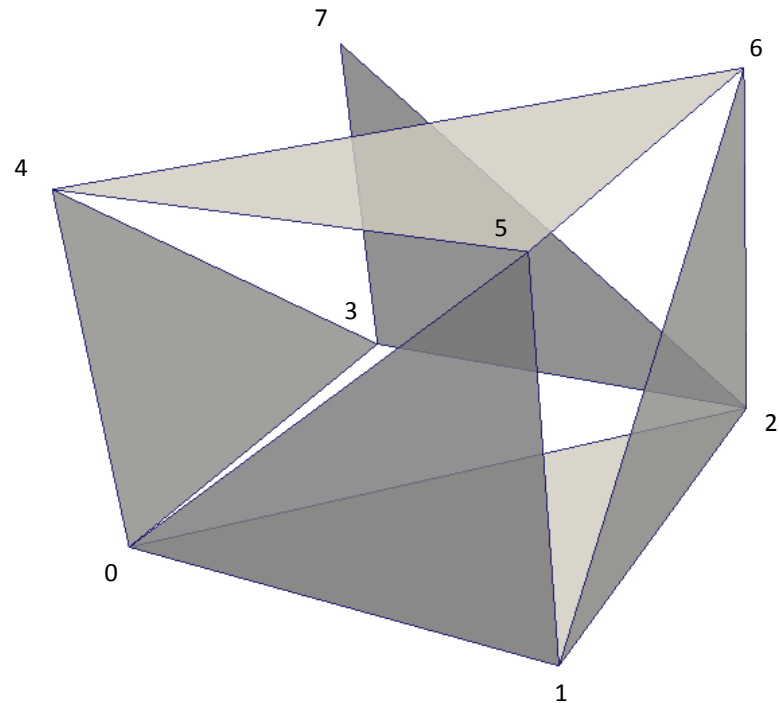
**\<drawtriangles\>**: draws a series of triangles defined by a set of points or a set of triangles.

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <drawfigure>
    - <points>
        <point x="0" y="0" z="0"/>
        <point x="1" y="0" z="0"/>
        <point x="1" y="1" z="0"/>
        <point x="0" y="1" z="0"/>
        <point x="0" y="0" z="0.8"/>
        <point x="1" y="0" z="0.8"/>
        <point x="1" y="1" z="0.8"/>
        <point x="0" y="1" z="0.8"/>
      </points>
    - <triangles>
        <triangle x="0" y="1" z="5"/>
        <triangle x="1" y="2" z="6"/>
        <triangle x="2" y="3" z="7"/>
        <triangle x="3" y="0" z="4"/>
        <triangle x="0" y="2" z="1"/>
        <triangle x="4" y="5" z="6"/>
      </triangles>
    </drawfigure>
</mainlist>
```
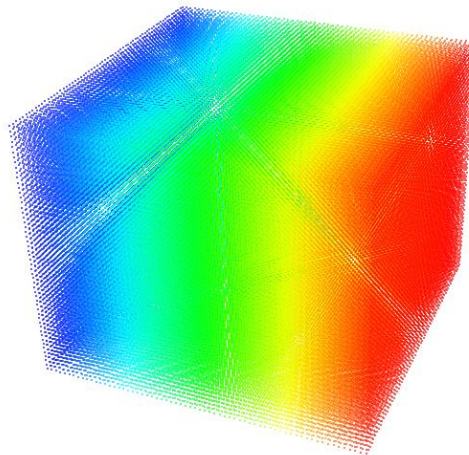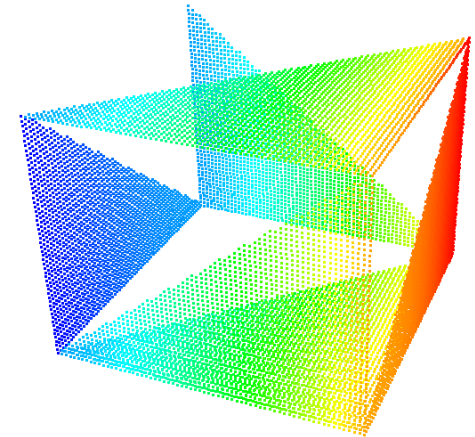
**<drawfigure>**: draws a solid figure made up of all the interior points to the planes formed by the given triangles



DrawFigure
(*drawmode=solid*)

DrawTriangles  or
DrawFigure  (*drawmode=face*)

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setdrawmode mode="full"/>
    <setmkbound mk="0"/>
  - <drawpyramid mask="0">
      <point x="0.25" y="0.25" z="0.7"/>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0" y="1" z="0"/>
    </drawpyramid>
    <shapeout file="Pyramid1" reset="true"/>
  - <drawpyramid mask="2">
      <point x="0.25" y="0.25" z="0.7"/>
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0" y="1" z="0"/>
    </drawpyramid>
    <shapeout file="Pyramid2" reset="true"/>
</mainlist>
```
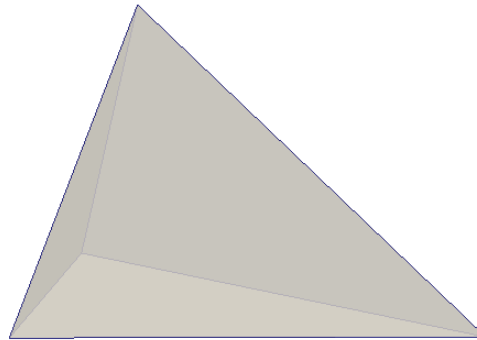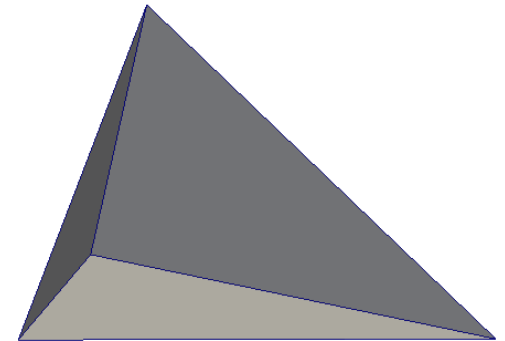
**<drawpyramid>**: draws a pyramid with the top point and other points of the base (minimum 3)

**mask** indicates the faces to be hidden with a bit to 1 the first bit corresponds always to the base and the rest to the faces following the order

Pyramid1
(*mask=0*)

Pyramid2
(*mask=2=0..010*)

```xml
- <mainlist>
    <setshapemode>dp|bound</setshapemode>
    <setdrawmode mode="full"/>
    <setmkbound mk="0"/>
  - <drawprism mask="0">
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0" y="1" z="0"/>
      <point x="0" y="0" z="0.5"/>
      <point x="1" y="0" z="0.5"/>
      <point x="0" y="1" z="0.5"/>
    </drawprism>
    <shapeout file="Prism1" reset="true"/>
  - <drawprism mask="2">
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0" y="1" z="0"/>
      <point x="0" y="0" z="0.5"/>
      <point x="1" y="0" z="0.5"/>
      <point x="0" y="1" z="0.5"/>
    </drawprism>
    <shapeout file="Prism2" reset="true"/>
</mainlist>
```

**<drawprism>**: draws a prism with a minimum of 6 points. the first half of points are the base and the second half the top (the number of points must be even)

**mask** indicates the faces to be hidden with a bit to 1
the first bit corresponds to the base, the second to the top and the rest to the faces following the order

Prism1 (*mask=0*)          Prism2 (*mask=2=0..010*)

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setdrawmode mode="full"/>
    <setmkbound mk="0"/>
  - <drawprism mask="0">
      <point x="0" y="0" z="0"/>
      <point x="1" y="-3" z="0"/>
      <point x="2" y="0" z="0"/>
      <point x="5" y="1" z="0"/>
      <point x="2" y="2" z="0"/>
      <point x="1" y="5" z="0"/>
      <point x="0" y="2" z="0"/>
      <point x="-3" y="1" z="0"/>
      <point x="0" y="0" z="6"/>
      <point x="1" y="-3" z="6"/>
      <point x="2" y="0" z="6"/>
      <point x="5" y="1" z="6"/>
      <point x="2" y="2" z="6"/>
      <point x="1" y="5" z="6"/>
      <point x="0" y="2" z="6"/>
      <point x="-3" y="1" z="6"/>
    </drawprism>
    <shapeout file="Prism3" reset="true"/>
  </mainlist>
```
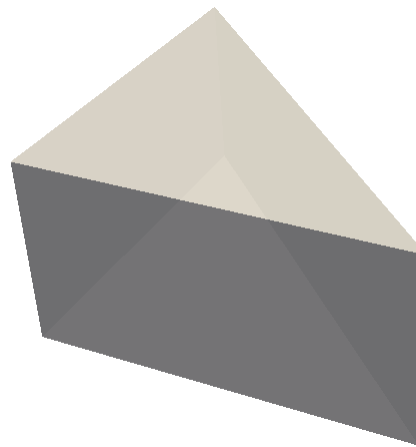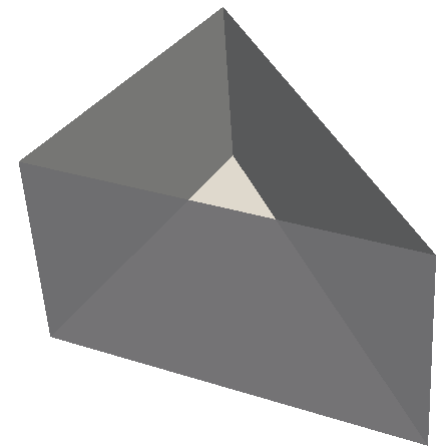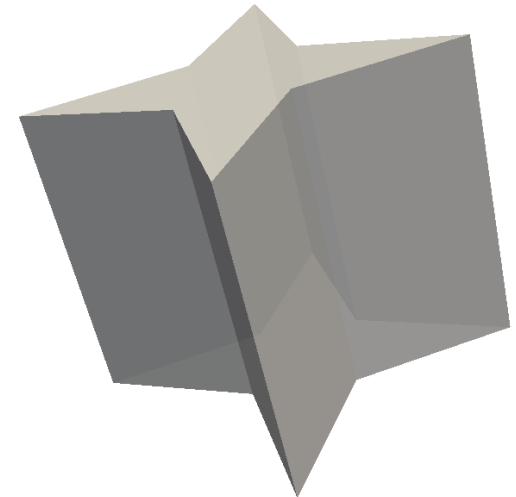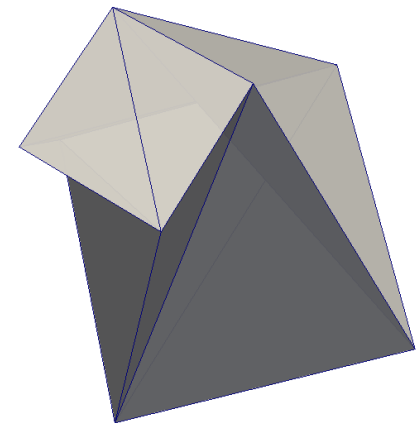
```
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setdrawmode mode="full"/>
    <setmkbound mk="0"/>
  - <drawprism mask="0">
      <point x="0" y="0" z="0"/>
      <point x="4" y="0" z="0"/>
      <point x="4" y="4" z="0"/>
      <point x="0" y="4" z="0"/>
      <point x="2" y="1" z="5"/>
      <point x="3" y="2" z="5"/>
      <point x="2" y="3" z="5"/>
      <point x="1" y="2" z="5"/>
    </drawprism>
    <shapeout file="Prism4" reset="true"/>
  </mainlist>
```
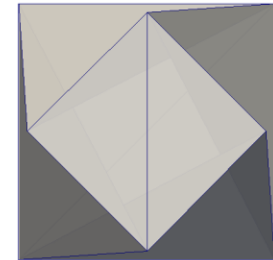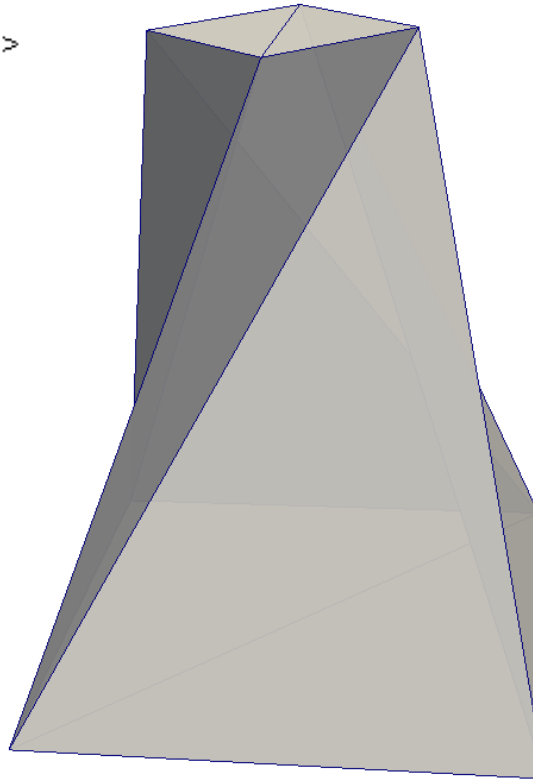
## EXAMPLE: *CaseWavemaker_Def.xml*

```xml
<setmkbound mk="0" />
<drawprism mask="1 | 2 | 6 | 7">
    <point x="5" y="0" z="1.5" />
    <point x="5" y="0" z="1.1" />
    <point x="1" y="0" z="0" />
    <point x="0" y="0" z="0" />
    <point x="0" y="0" z="1.5" />
    <point x="5" y="2" z="1.5" />
    <point x="5" y="2" z="1.1" />
    <point x="1" y="2" z="0" />
    <point x="0" y="2" z="0" />
    <point x="0" y="2" z="1.5" />
</drawprism>
```
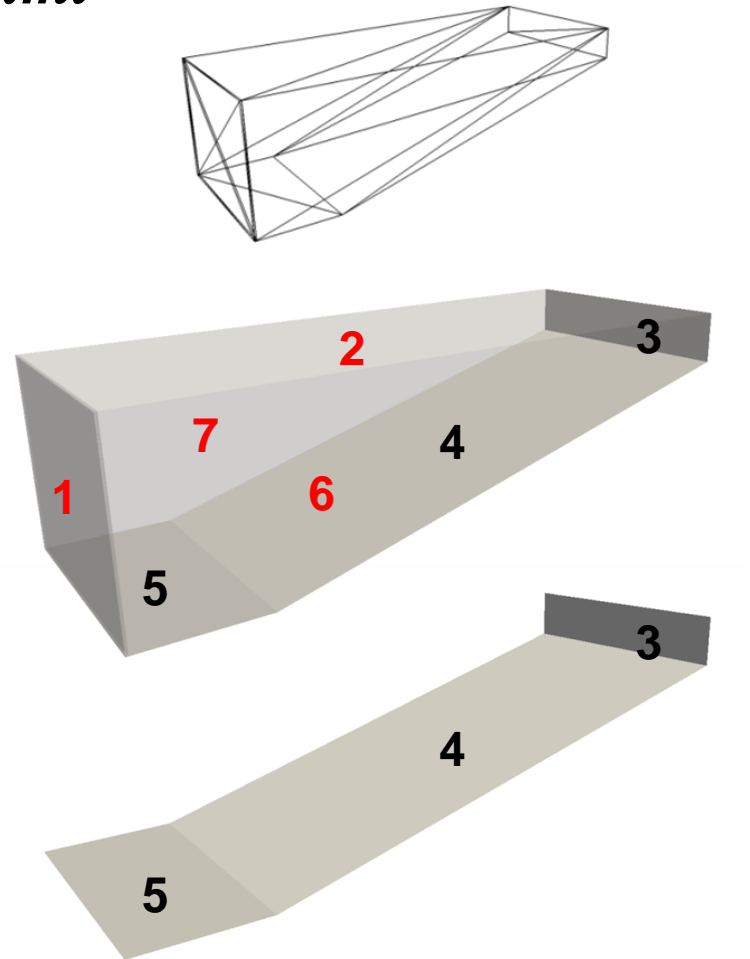
**mask** in **<drawprism>** are the faces to be hidden

and can be also defined using the index of the faces instead of bits

In this example, faces 1, 2, 6 and 7 are not created, only 3,4 and 5.

```xml
- <mainlist>
    <setshapemode>dp | bound</setshapemode>
    <setmkbound mk="0"/>
  - <drawbox>
      <boxfill>solid</boxfill>
      <point x="0" y="0" z="0"/>
      <size x="1" y="1" z="0.5"/>
    </drawbox>
    <shapeout file="BoxSolid" reset="true"/>
  - <drawbox>
      <boxfill>all</boxfill>
      <point x="0" y="0" z="0"/>
      <size x="1" y="1" z="0.5"/>
    </drawbox>
    <shapeout file="BoxA" reset="true"/>
  - <drawbox>
      <boxfill>all ^ top</boxfill>
      <point x="0" y="0" z="0"/>
      <size x="1" y="1" z="0.5"/>
    </drawbox>
    <shapeout file="BoxB" reset="true"/>
  - <drawbox>
      <boxfill>bottom | left | right</boxfill>
      <point x="0" y="0" z="0"/>
      <size x="1" y="1" z="0.5"/>
    </drawbox>
    <shapeout file="BoxC" reset="true"/>
</mainlist>
```
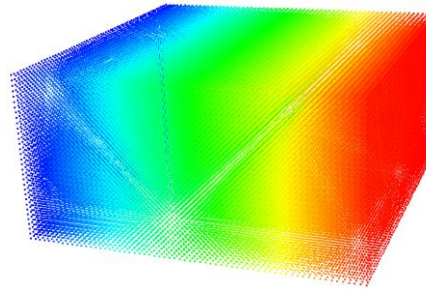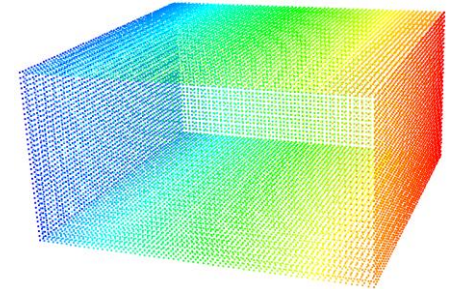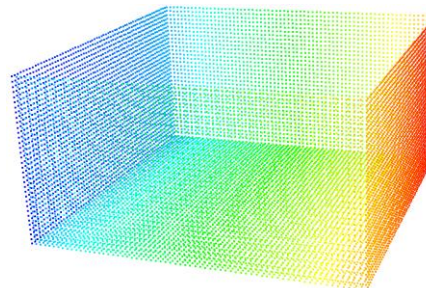
**<drawbox>**: draws a box with an initial point and the size
**<boxfill>** indicates if *solid* or *face* and what faces are not hide



BoxSolid (solid)

BoxA  (all)

BoxB (all^top)

BoxC  (bottom|right|right)

27
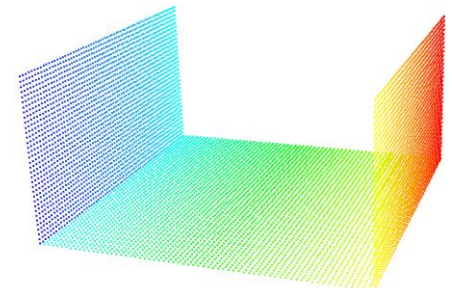
```
- <mainlist>
    <setshapemode>dp | solid</setshapemode>
    <setmkbound mk="10"/>
    <setdrawmode mode="face"/>
  - <drawbeach mask="128">
      <point x="0" y="3" z="1.2"/>
      <point x="0" y="3" z="0"/>
      <point x="1.7" y="3" z="0"/>
      <point x="3.5" y="3" z="0.5"/>
      <point x="5.5" y="3" z="0.5"/>
      <point x="5.5" y="3" z="1.2"/>
    </drawbeach>
  </mainlist>
```
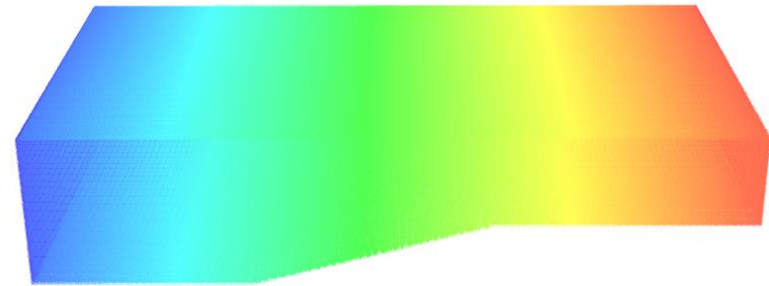


BeachFace



BeachFace
(mask="128")

**<drawbeach>**: draws a beach with the lateral points that formed the profile of the beach
**mask** indicates the faces to be hidden .

```
– <mainlist>
    <setmkbound mk="0"/>
    <setdrawmode mode="solid"/>
  – <drawsphere radius="0.8">
      <point x="1" y="1" z="1"/>
    </drawsphere>
    <setdrawmode mode="face"/>
  – <drawsphere radius="0.8">
      <point x="1" y="1" z="1"/>
    </drawsphere>
</mainlist>
```
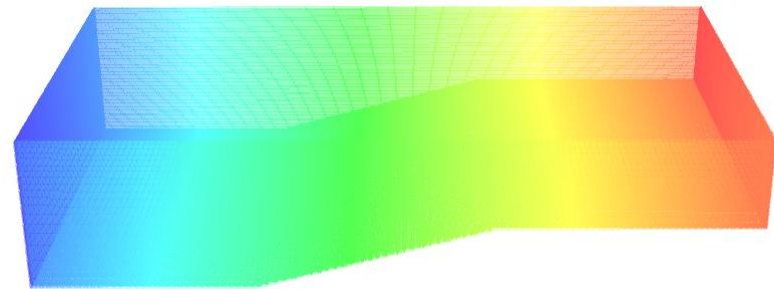
**<drawsphere>**: draws a sphere with the center point and the radius
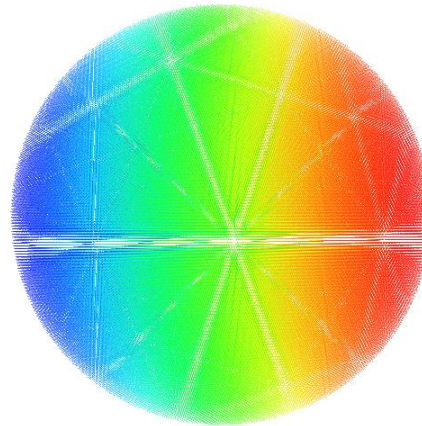
Sphere
(*drawmode=solid*)
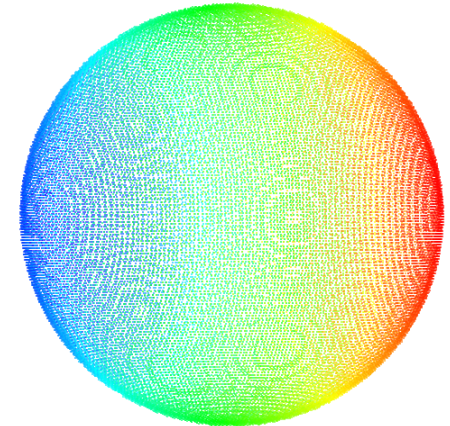
Sphere
(*drawmode=face*)

```
<setdpctes ctesphere="0.4"/>
<setdpctes ctespherenumsides="40"/>
```

when *face:*

**ctesphere** indicates the width of the sphere

**ctespherenumsides** indicates the number of triangles used to create the VTK of polygons

**<drawcylinder>**: draws a cylinder with two points and radius

**mask** indicates the faces to be hide

```
– <mainlist>
    <setmkbound mk="0"/>
    <setdrawmode mode="solid"/>
  – <drawcylinder radius="1" mask="0">
      <point  x="1.5" y="1.5" z="0.5"/>
      <point  x="1.5" y="1.5" z="3"/>
    </drawcylinder>
    <setdrawmode mode="face"/>
  – <drawcylinder radius="1" mask="0">
      <point  x="3.5" y="2.5" z="0.5"/>
      <point  x="3.5" y="2.5" z="3"/>
    </drawcylinder>
  </mainlist>
```



Cylinder
(*drawmode=solid*)



Cylinder
(*drawmode=face*)

```
<setdpctes ctecylindertube="0.6"/>
<setdpctes ctecylindercover="0.7"/>
<setdpctes ctecylindernumsides="40"/>
```

when *face:*

**ctecylindertube** indicates the width of the tube

**ctecylindercover** indicates the width of the covers

**ctespherenumsides** indicates the number of triangles used to create the VTK of polygons

# CASEDEF-GEOMETRY-COMMANDS-MAINLIST

```xml
- <mainlist>
    <setmkbound mk="0"/>
    <drawfilestl file="File.stl"/>
    <drawfileply file="File.ply"/>
    <drawfileply file="File.vtk"/>
    - <drawfilestl file="File.stl">
        <drawmove x="0.5" y="0" z="0"/>
        <drawrotate angx="10" angy="15" angz="30"/>
        <drawscale x="1" y="1" z="0.8"/>
    </drawfilestl>
    - <drawfileply file="File.ply">
        <drawmove x="0.5" y="0" z="0"/>
    </drawfileply>
    - <drawfileply file="File.ply">
        <drawmove x="0.5" y="0" z="0"/>
        <drawrotate angx="10" angy="15" angz="30"/>
    </drawfileply>
    - <drawfileply file="File.ply">
        <drawrotate angx="10" angy="15" angz="30"/>
    </drawfileply>
    - <drawfilevtk file="File.vtk">
        <polyselec>points</polyselec>
    </drawfilevtk>
    - <drawfilevtk file="File.vtk">
        <polyselec>points | lines</polyselec>
    </drawfilevtk>
    - <drawfilevtk file="File.vtk">
        <polyselec>triangles</polyselec>
    </drawfilevtk>
    - <drawfilevtk file="File.vtk">
        <polyselec>polygons</polyselec>
    </drawfilevtk>
</mainlist>
```
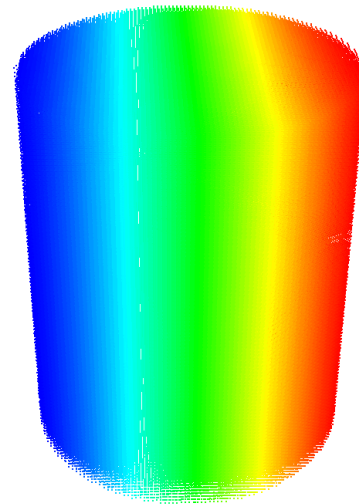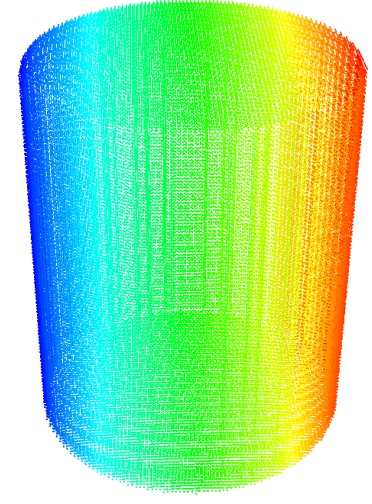
**<drawfilevtk>:** load a VTK file to be converted into points

**<drawfileply>:** load a PLY file to be converted into points

**<drawfilestl>:** load a STL file to be converted into points

Some modifications can be applied to the VTK, PLY or STL
**drawmove** a displacement is applied to the external object
**drawrotate** a rotation is applied to the external object
**drawscale** scaling is applied to the external object

31

```
- <mainlist>
    <setshapemode>real | bound | dp</setshapemode>
    <setmkbound mk="0"/>
    <drawfilevtk file="pump_fixed.vtk"/>
    <setmkbound mk="1"/>
    <drawfilevtk file="pump_moving.vtk"/>
    <setmkfluid mk="0"/>
  - <fillbox x="0.14" y="-0.1" z="-0.39">
      <modefill>void</modefill>
      <point x="-0.6" y="-0.39" z="-0.8"/>
      <size x="0.9" y="0.68" z="0.52"/>
    </fillbox>
  </mainlist>
```
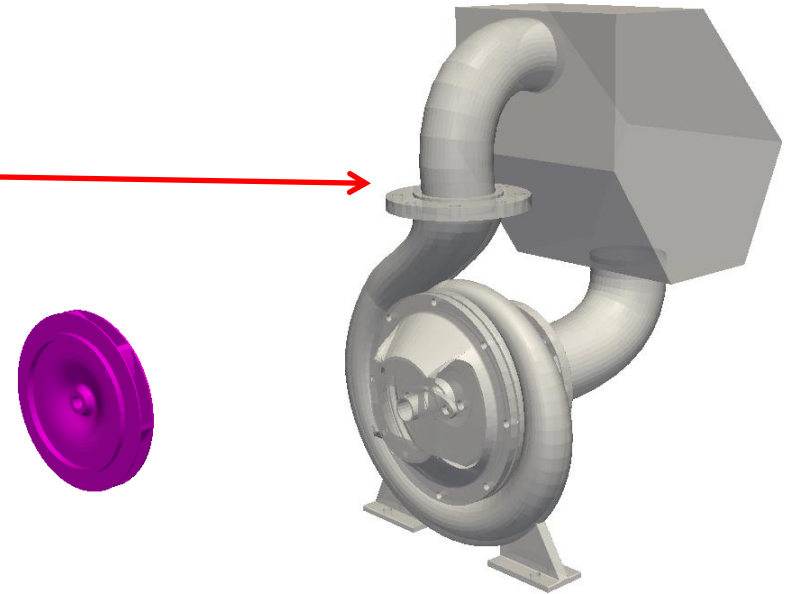
**from VTK to points**

```xml
- <mainlist>
    <setmkfluid mk="0"/>
    <fillvoidpoint x="3" y="2" z="1"/>
  - <fillpoint x="3" y="2" z="1">
      <modefill>void</modefill>
    </fillpoint>
  - <fillpoint x="1" y="1" z="1" mkfluid="0">
      <modefill>fluid</modefill>
    </fillpoint>
  - <fillpoint x="1" y="1" z="1" mkbound="0">
      <modefill>bound</modefill>
    </fillpoint>
  - <fillpoint x="2" y="2" z="2" mkfluid="2" mkbound="8">
      <modefill>border | void | fluid | bound</modefill>
    </fillpoint>
  - <fillbox x="0" y="1" z="0">
      <modefill>border</modefill>
      <point x="0.1" y="1" z="1.1"/>
      <size x="3" y="4" z="2"/>
    </fillbox>
  - <fillprism x="2" y="3" z="5">
      <point x="0" y="0" z="0"/>
      <point x="1" y="0" z="0"/>
      <point x="0" y="1" z="0"/>
      <point x="0" y="0" z="0.5"/>
      <point x="1" y="0" z="0.5"/>
      <point x="0" y="1" z="0.5"/>
      <modefill>void</modefill>
    </fillprism>
    <debugout/>
</mainlist>
```
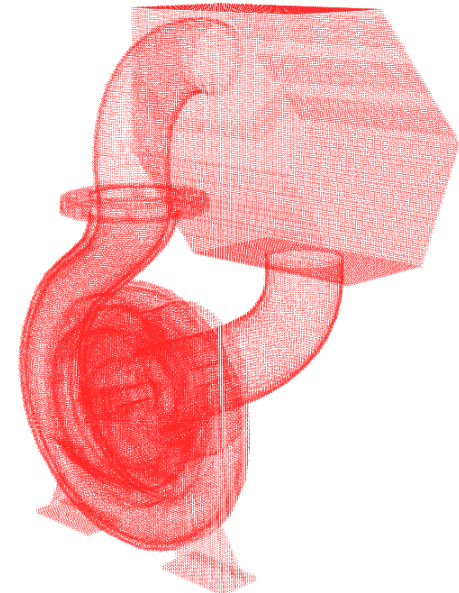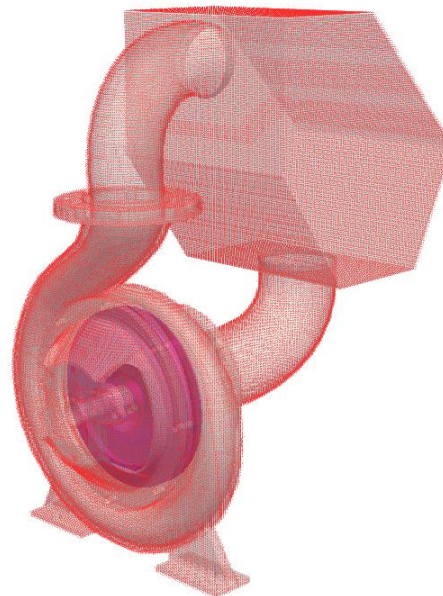
**<fillpoint>:** fills with points starting from the seed

**<fillbox>:** fills with points starting from the seed within the limits defined by a box

**<fillfigure>:** fills with points starting from the seed within the limits defined by a figure

**<fillprism>:** fills with points starting from the seed within the limits defined by a prism

**<modefill>** indicates what type of points can be filled with *void, fluid, bound*, it fills while a given type using *border* it fills until a given type

33

```
- <mainlist>
    <setshapemode>real | bound | dp</setshapemode>
    <setmkbound mk="0"/>
    <drawfilevtk file="pump_fixed.vtk"/>
    <setmkbound mk="1"/>
    <drawfilevtk file="pump_moving.vtk"/>
    <setmkfluid mk="0"/>
  - <fillbox x="0.14" y="-0.1" z="-0.39"
        <modefill>void</modefill>
        <point x="-0.6" y="-0.39" z="-0.8"/>
        <size x="0.9" y="0.68" z="0.52"/>
    </fillbox>
</mainlist>
```

**seed point**

**limits of the domain to fill**

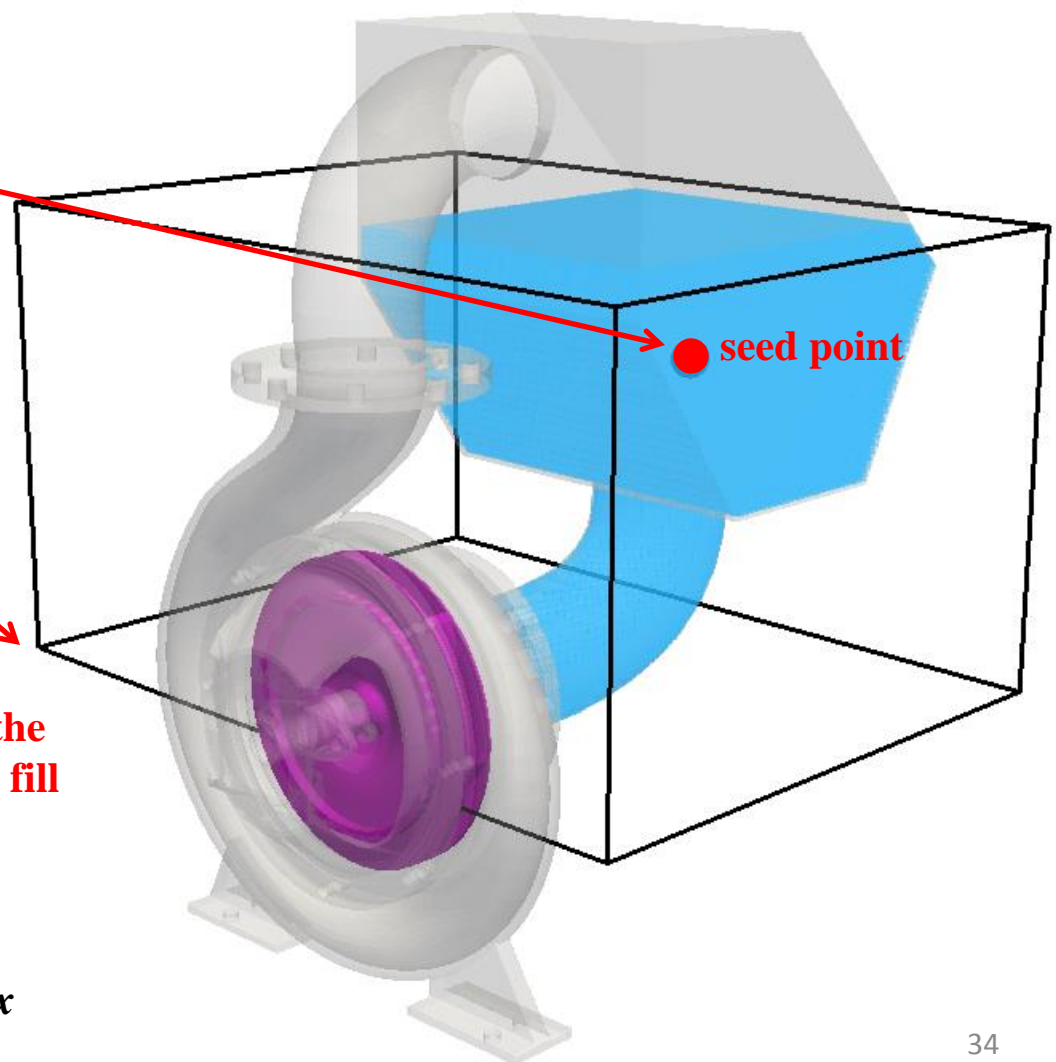**filling with *fluid* while *void* and before the limits defined by a *box***

```xml
- <geometry>
  - <definition dp="0.01">
      <pointmin x="-1" y="-0.05" z="-0.05"/>
      <pointmax x="2" y="1.1" z="2"/>
    </definition>
  - <commands>
    - <mainlist>
        <setshapemode>real | dp | bound</setshapemode>
        <setdrawmode mode="full"/>
        <setmkfluid mk="1"/>
      - <drawsphere radius="0.15">
          <point x="-0.55" y="0.5" z="0.18"/>
        </drawsphere>
        <setmkfluid mk="2"/>
      - <drawbox>
          <boxfill>solid</boxfill>
          <point x="1.4" y="0.35" z="0.01"/>
          <size x="0.3" y="0.3" z="0.3"/>
        </drawbox>
      </mainlist>
    </commands>
  </geometry>
- <initials>
    <velocity mkfluid="1" x="1.05" y="0" z="4.905"/>
    <velocity mkfluid="2" x="-0.875" y="0" z="5.886"/>
  </initials>
```

**<initials>:** special behaviours can be imposed to a set of fluid particles labeled with a *mk*, such as;

**<velocity>** initial velocity defined by a vector

**<velwave>** a solitary wave defined by *depth* and *amplitude*

```
- <geometry>
    - <definition dp="0.01">
        <pointmin x="-1" y="-0.05" z="-0.05"/>
        <pointmax x="2" y="1.1" z="2"/>
    </definition>
    - <commands>
        - <mainlist>
            <setshapemode>real | dp | bound</setshapemode>
            <setdrawmode mode="full"/>
            <setmkfluid mk="1"/>
            - <drawsphere radius="0.15">
                <point x="-0.55" y="0.5" z="0.18"/>
            </drawsphere>
            <setmkfluid mk="2"/>
            - <drawbox>
                <boxfill>solid</boxfill>
                <point x="1.4" y="0.35" z="0.01"/>
                <size x="0.3" y="0.3" z="0.3"/>
            </drawbox>
        </mainlist>
    </commands>
</geometry>
- <initials>
    <velocity mkfluid="1" x="1.05" y="0" z="4.905"/>
    <velocity mkfluid="2" x="-0.875" y="0" z="5.886"/>
</initials>
```
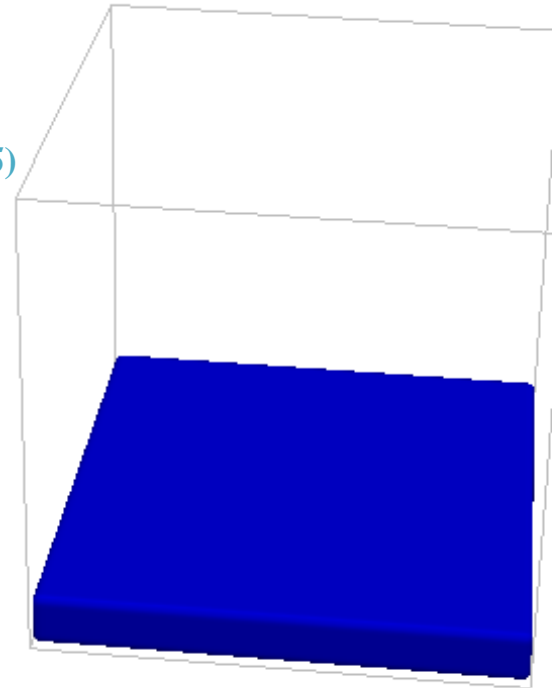
**different initial velocities are imposed to two volumes of fluid *mk=1* (sphere) and *mk=2* (box)**

$v_{sphere} = (1.05, 0, 4.905)$

$v_{box} = (-0.875, 0, 5.886)$

```
− <initials>
    <velocity mkfluid="1" x="1.05" y="0" z="4.905"/>
    <velocity mkfluid="2" x="-0.875" y="0" z="5.886"/>
  </initials>
```
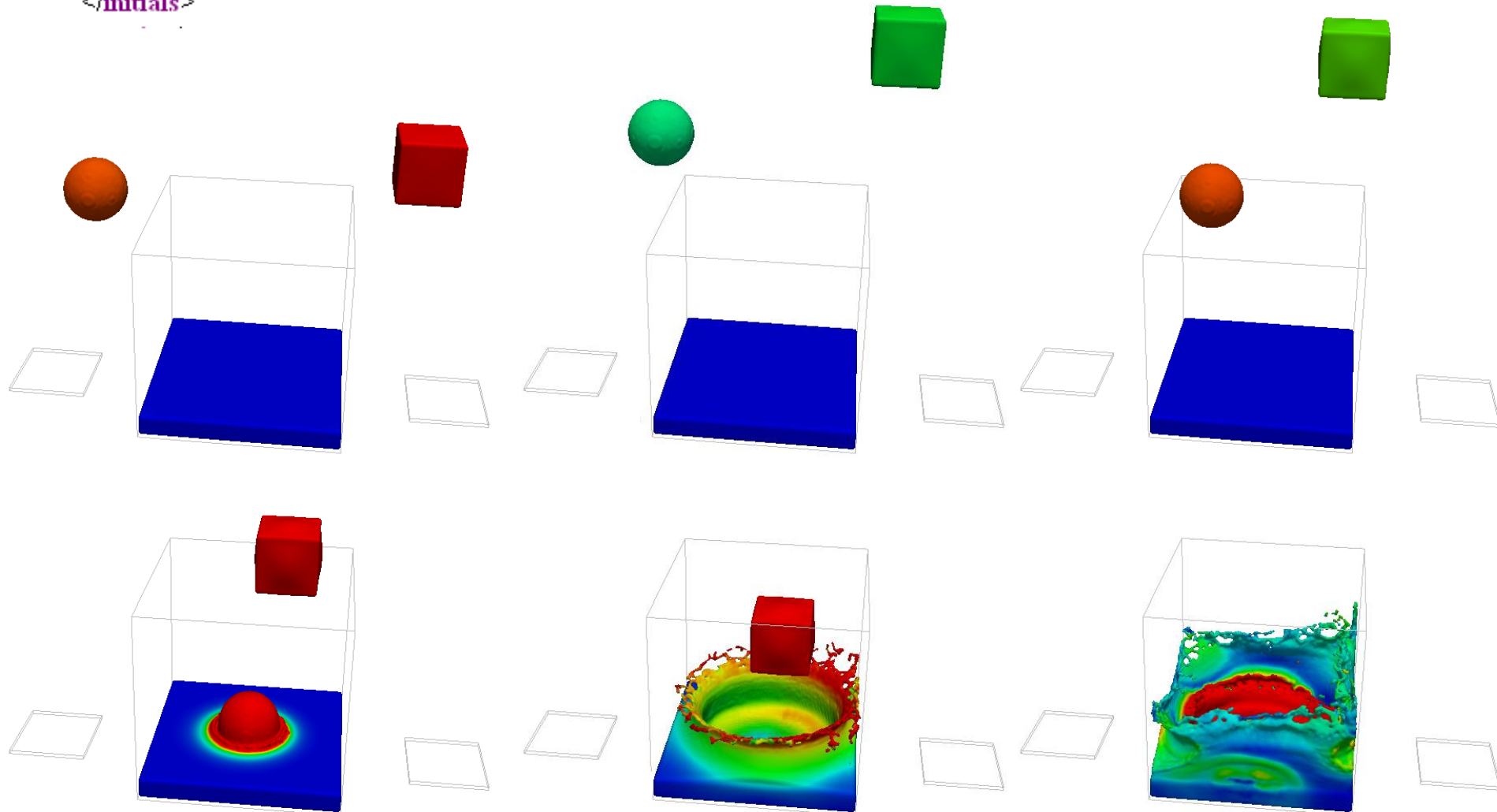
**colour represents velocity**

```
− <floatings>
    <floating mkbound="0" relativeweight="1.3"/>
  − <floating mkbound="1" relativeweight="1.3">
      <velini x="1" y="3" z="2"/>
      <omegaini x="0.2" y="0.4" z="0.6"/>
    </floating>
  − <floating mkbound="2">
      <massbody value="1300"/>
      <center x="11" y="12" z="13"/>
      <inertia x="20" y="22" z="24"/>
    </floating>
  − <floating mkbound="3">
      <massbody value="1300"/>
      <center x="11" y="12" z="13"/>
      <inertia x="20" y="22" z="24"/>
      <velini x="1" y="3" z="2"/>
      <omegaini x="0.2" y="0.4" z="0.6"/>
    </floating>
  − <floating mkbound="4">
      <massbody value="1300"/>
      <inertia x="20" y="22" z="24"/>
    </floating>
  </floatings>
```

**<floatings>:** indicates for a set of particles labeled with a *mk* that these particles constitute a floating object

**massbody** total mass of the object
**center** gravity center of the rigid object
**inertia** momentum of inertia of the rigid object
**velini** initial linear velocity of the object
**omegaini** initial angular velocity of the object

these variables are computed by GenCase or can be specified

# CASEDEF-PROPERTIES

**<properties>:** New properties can be defined and will be assigned to one or several *mk*.

```
<casedef>
    ...
    <properties>
        <links>
            <link mkfluid="0" property="material_1" />
            <link mkbound="3-6,1" property="material_2+data_x" />
        </links>
        <propertyfile file="run/ftdata_ext.xml" path="case.materials" />
        <property name="material_1" weight="1.35" other="pepe"/>
        <property name="material_2" begin="168" count="973">
            <massbody value="4728.78" />
            <center x="4.99" y="5" z="7.03" />
        </property>
        <property name="data_x" weight="1.35" />
    </properties>
</casedef>
```

Each label of **property** has a name and can group multiple values whose content can be text or number, such as *weight* and *other* in

```
<property name="material_1" weight="1.35" other="pepe"/>
```

**property** can also contain values with subvalues such as *massbody* and *center* in:

```
<property name="material_2" begin="168" count="973">
    <massbody value="4728.78" />
    <center x="4.99" y="5" z="7.03" />
</property>
```

These properties can be also loaded from an external file using **propertyfile**
The name of the file and the path to access to the section with properties must be indicated

```
<propertyfile file="run/ftdata_ext.xml" path="case.materials" />
```

Example of the file *ftdata_ext.xml*:

```
<case>
    <materials>
        <property name="uno" value="1.35"/>
        <property name="dos" value="168">
            <massbody value="4728.78" />
        </property>
    </materials>
</case>
```

# CASEDEF-PROPERTIES

Section **links:** one or more **property** can be assigned with one or several values of *mk*

In `<link mkfluid="0" property="material_1"/>`

values of *material_1* are assigned to fluid particles with mk=0.

In `<link mkbound="3-6,1" property="material_2+data_x"/>`

values of *material_2* and *data_x* are assigned to boundary particles with mk 1, 3, 4, 5 and 6.

A **property** can be also directly indicated in the definition of the floating bodies:

```
<floatings>
    ...
    <floating mkbound="4" property="Material_2">
        <massbody value="1300" />
        <inertia x="20" y="22" z="24" />
    </floating>
</floatings>
```

# CASEDEF-PROPERTIES

**GenCase** loads the information of **case.casedef.properties** and writes in **case.execution.particles.properties** what will be used by DualSPHysics (only the properies that will be used and only the mk values that exist)

```
<particles np="1494" nb="313" nbf="313" mkboundfirst="11" mkfluidfirst="1">
    <fixed mkbound="0" mk="11" begin="0" count="229" />
    <fixed mkbound="1" mk="12" begin="229" count="28" property="data_x+material_2" />
    <fixed mkbound="2" mk="13" begin="257" count="28" />
    <fixed mkbound="4" mk="15" begin="285" count="28" property="data_x+material_2" />
    <fluid mkfluid="0" mk="1" begin="313" count="1146" property="material_1" />
    <fluid mkfluid="1" mk="2" begin="1459" count="35" />
    <properties>
        <links>
            <link mk="1" property="material_1" />
            <link mk="12,15" property="data_x+material_2" />
        </links>
        <property name="material_1" weight="1.35" other="pepe"/>
        <property name="material_2" begin="168" count="973">
            <massbody value="4728.78" />
            <center x="4.99" y="5" z="7.03" />
        </property>
        <property name="data_x" weight="1.35" />
    </properties>
</particles>
```

•*Motion01*: uniform rectilinear motion (**<mvrect />**) that also includes pauses (**<wait />**)

```xml
- <motion>
  - <objreal ref="1">
      <begin mov="1" start="0" finish="5.4"/>
    - <mvrect id="1" duration="0.6" next="2">
        <vel x="1" y="0" z="0"/>
      </mvrect>
      <wait id="2" duration="0.3" next="3"/>
    - <mvrect id="3" duration="0.6" next="4">
        <vel x="1" y="0" z="0"/>
      </mvrect>
      <wait id="4" duration="0.3" next="5"/>
    - <mvrect id="5" duration="0.6" next="6">
        <vel x="1" y="0" z="0"/>
      </mvrect>
      <wait id="6" duration="0.3" next="7"/>
    - <mvrect id="7" duration="-1" next="1">
        <vel x="-1.8" y="0" z="0"/>
      </mvrect>
    </objreal>
  </motion>
```
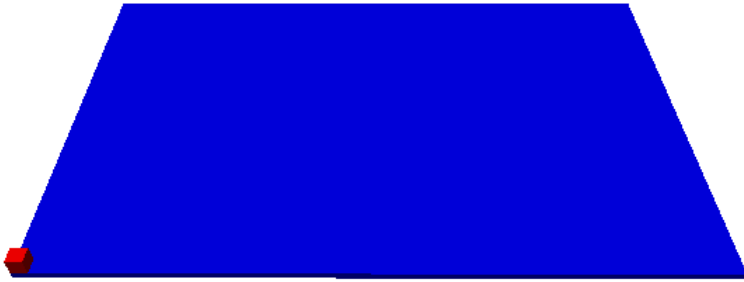
**movement defined for the set of particles with *mk=1***

**first mov=1 during 0.6s,
then wait=2 for 0.3s,
then mov=3 during 0.6s,
then wait=4 for 0.3s,
then mov=5 during 0.6s…**

**<mvrect>:** uniform rectilinear movement

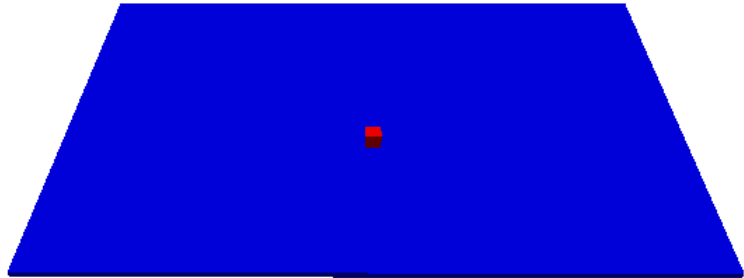**vel** indicates the constant velocity vector

# CASEDEF-MOTION

•*Motion01*: uniform rectilinear motion (**<mvrect />**) that also includes pauses (**<wait />**)

Time: 0.00 s

Time: 0.30 s

Time: 0.60 s

Time: 0.90 s

Time: 1.20 s

Time: 1.50 s

Time: 1.80 s

Time: 2.10 s

Time: 2.40 s

Time: 2.70 s

# CASEDEF-MOTION

•*Motion02*: combination of two uniform rectilinear motion (**<mvrect />**)

```xml
− <motion>
  − <objreal ref="1">
      <begin mov="1" start="0"/>
    − <mvrect id="1" duration="2" next="2">
        <vel x="1" y="0" z="0"/>
      </mvrect>
    − <mvrect id="2" duration="1" next="1">
        <vel x="-2" y="0" z="0"/>
      </mvrect>
      <begin mov="3" start="0.5"/>
    − <mvrect id="3" duration="1.3" next="4">
        <vel x="0" y="1" z="0"/>
      </mvrect>
    − <mvrect id="4" duration="1.3" next="3">
        <vel x="0" y="-1" z="0"/>
      </mvrect>
    </objreal>
  </motion>
```
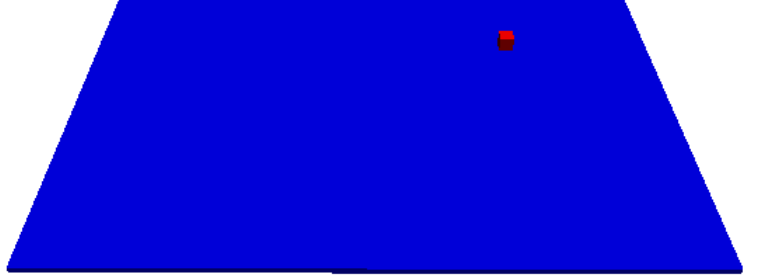
**<mvrect>:** uniform rectilinear movement

**vel** indicates the constant velocity vector

# CASEDEF-MOTION

•*Motion02*: combination of two uniform rectilinear motion (**<mvrect />**)



Time:   0.00 s
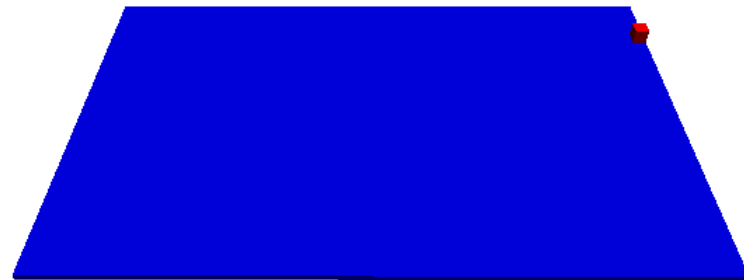
Time:   0.50 s

Time:   1.00 s

Time:   1.50 s

Time:   2.00 s

Time:   2.50 s

```xml
- <motion>
  - <objreal ref="1">
      <begin mov="1" start="0"/>
    - <mvrect id="1" duration="1.5" next="2">
        <vel x="1" y="0" z="0"/>
      </mvrect>
    - <mvrect id="2" duration="1.5" next="1">
        <vel x="-1" y="0" z="0"/>
      </mvrect>
      <begin mov="3" start="0.1"/>
    - <mvrect id="3" duration="1.1" next="4">
        <vel x="0" y="1" z="0"/>
      </mvrect>
    - <mvrect id="4" duration="1.1" next="3">
        <vel x="0" y="-1" z="0"/>
      </mvrect>
  - <objreal ref="2">
      <begin mov="1" start="0.2"/>
    - <mvrect id="1" duration="0.45" next="2">
        <vel x="1" y="0" z="0"/>
      </mvrect>
    - <mvrect id="2" duration="0.45" next="3">
        <vel x="0" y="1" z="0"/>
      </mvrect>
    - <mvrect id="3" duration="0.45" next="4">
        <vel x="-1" y="0" z="0"/>
      </mvrect>
    - <mvrect id="4" duration="0.45" next="1">
        <vel x="0" y="-1" z="0"/>
      </mvrect>
    </objreal>
  </objreal>
</motion>
```
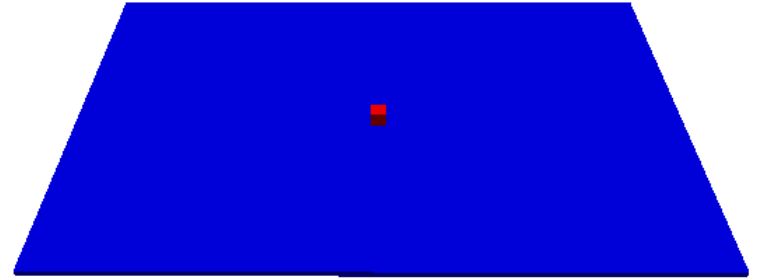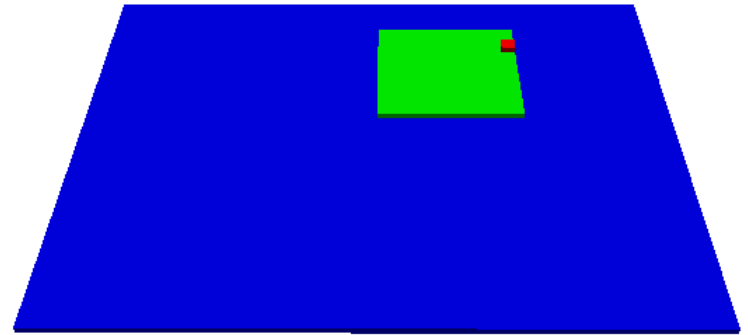
• *Motion03*: movement of an object depending on the movement of another (hierarchy of objects)

**movement defined for the set of particles with *mk=2* that also moves according to the movement defined for *mk=1***
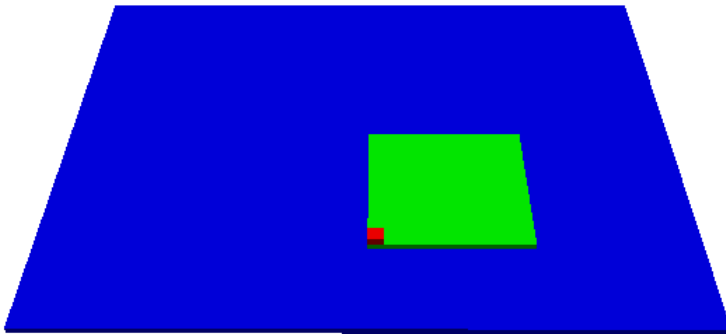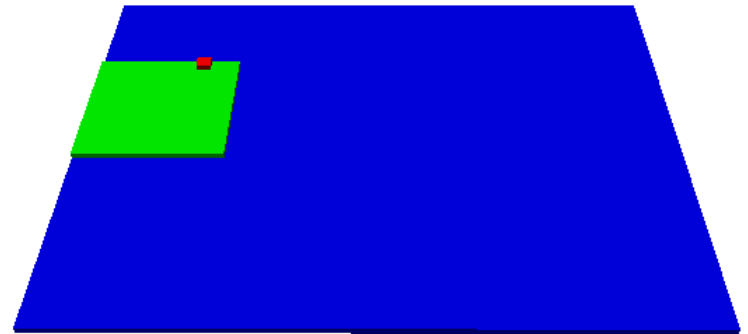
47

•*Motion03*: movement of an object depending on the movement of another (hierarchy of objects)



Time:   0.00 s

Time:   1.00 s

Time:   2.00 s

Time:   3.00 s

# CASEDEF-MOTION

•*Motion04*: accelerated rectilinear motion (**<mvrectace />**)

```xml
- <motion>
  - <objreal ref="1">
      <begin mov="1" start="0"/>
    - <mvrectace id="1" duration="1.411" next="2">
        <velini x="0" y="0" z="0"/>
        <ace x="2" y="0" z="0"/>
      </mvrectace>
    - <mvrectace id="2" duration="1">
        <velini x="-2" y="5" z="0"/>
        <ace x="0" y="-10" z="0"/>
      </mvrectace>
    </objreal>
  </motion>
```

**<mvrectace>:** accelerated rectilinear movement

**velini** indicates the initial velocity vector
**ace** indicates the acceleration vector

# CASEDEF-MOTION

•*Motion04*: accelerated rectilinear motion (**<mvrectace />**)

Time:   0.00 s

Time:   0.75 s

Time:   1.50 s

Time:   2.25 s

•*Motion05*: rotational motion (**<mvrot />**)

```
- <motion>
  - <objreal ref="3">
      <begin mov="1" start="0"/>
    - <mvrot id="1" duration="1000">
        <vel ang="20"/>
        <axisp1 x="0.5" y="0.5" z="0"/>
        <axisp2 x="0.5" y="0.5" z="1"/>
      </mvrot>
    </objreal>
  - <objreal ref="4">
      <begin mov="1" start="0"/>
    - <mvrot id="1" duration="1000">
        <vel ang="240"/>
        <axisp1 x="0.5" y="0.5" z="0"/>
        <axisp2 x="0.5" y="0.5" z="1"/>
      </mvrot>
    </objreal>
  </motion>
```
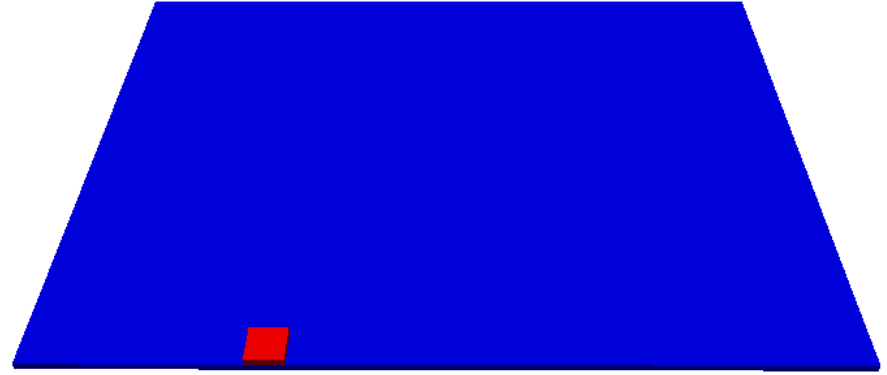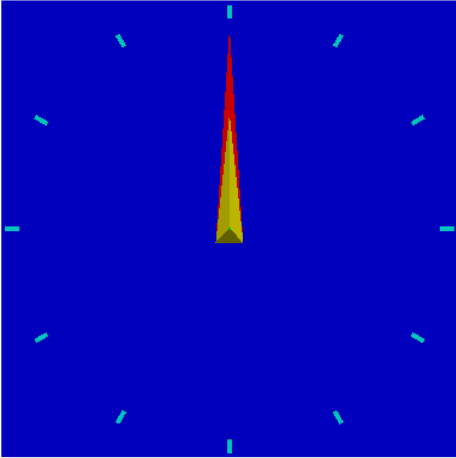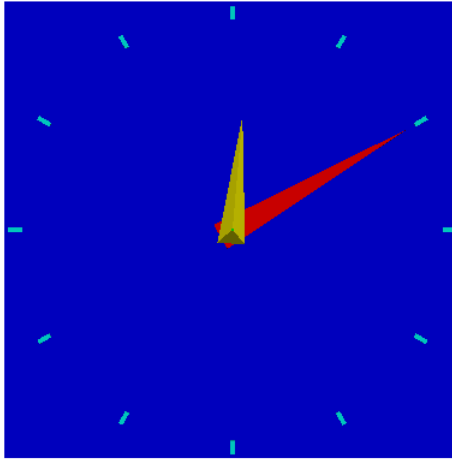
**<mvrot>:** rotational movement

**vel** indicates the angular velocity
**axisp1** first point of the rotation axis
**axisp2** second point of the rotation axis
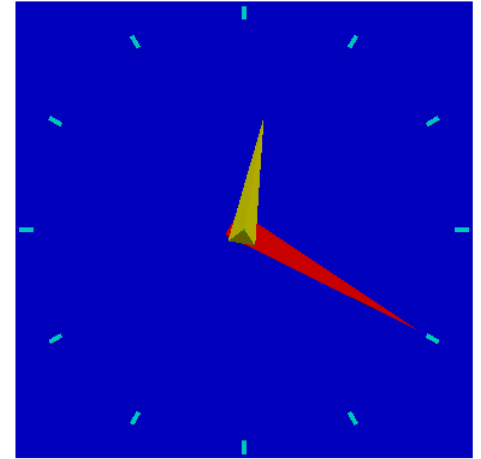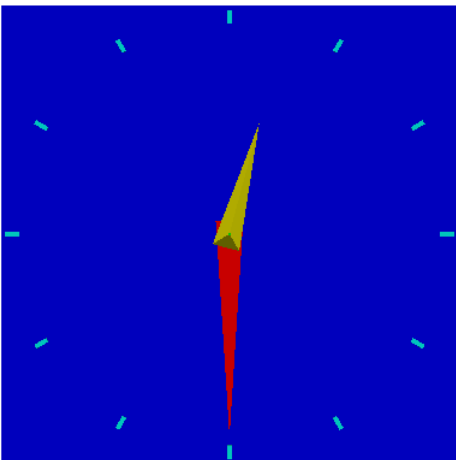
# CASEDEF-MOTION

•*Motion05*: rotational motion (**<mvrot />**)
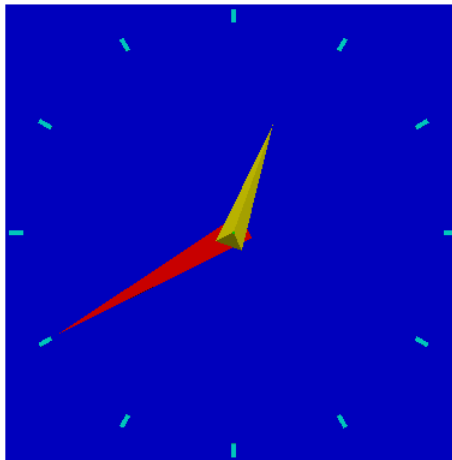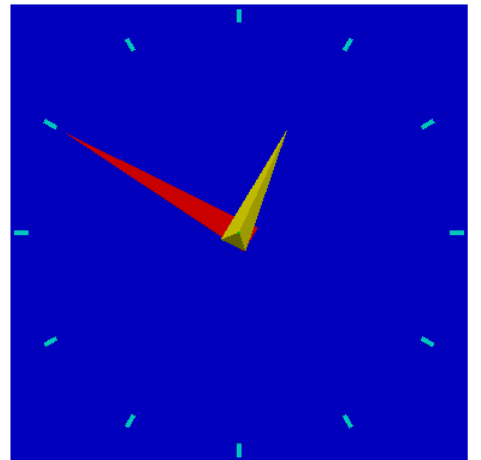


Time:   0.00 s

Time:   0.25 s

Time:   0.50 s

Time:   0.75 s

Time:   1.00 s

Time:   1.25 s

•*Motion06*: accelerated rotation motion (**&lt;mvrotace /&gt;**) and accelerated circular motion (**&lt;mvcirace /&gt;**).

```
− <motion>
  − <obj>
      <objreal ref="1"/>
      <objreal ref="3"/>
      <objreal ref="4"/>
      <begin mov="1" start="0"/>
    − <mvrotace id="1" duration="1000">
        <ace ang="9"/>
        <velini ang="-50"/>
        <axisp1 x="0" y="0" z="1.85"/>
        <axisp2 x="0" y="1" z="1.85"/>
      </mvrotace>
    </obj>
  − <objreal ref="5">
      <begin mov="1" start="0"/>
    − <mvcirace id="1" duration="1000">
        <ace ang="9"/>
        <velini ang="-50"/>
        <ref x="1.3" y="-0.7" z="1.85"/>
        <axisp1 x="0" y="0" z="1.85"/>
        <axisp2 x="0" y="1" z="1.85"/>
      </mvcirace>
    </objreal>
  </motion>
```

**&lt;mvrotace&gt;:** accelerated rotational movement

**ace** indicates the angular acceleration
**velini** indicates the initial angular velocity
**axisp1** first point of the rotation axis
**axisp2** second point of the rotation axis

**&lt;mvcirace&gt;:** accelerated circular movement

**ace** indicates the angular acceleration
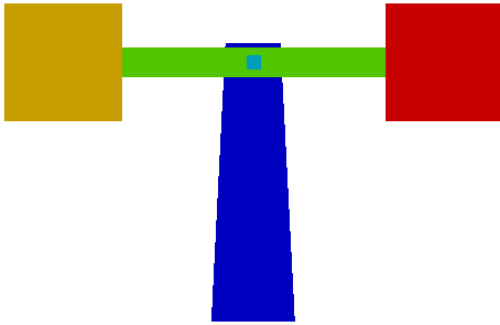**ref** indicates the point of the object that rotates with the axis
**velini** indicates the initial angular velocity
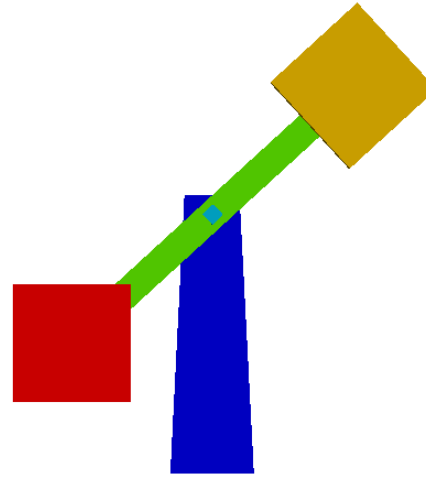**axisp1** first point of the rotation axis
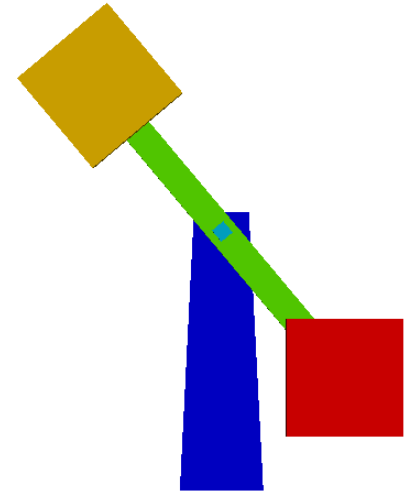**axisp2** second point of the rotation axis

# CASEDEF-MOTION

•*Motion06*: accelerated rotation motion (**&lt;mvrotace /&gt;**) and accelerated circular motion (**&lt;mvcirace /&gt;**).



Time:   0.00 s

Time:   5.00 s

Time:   10.00 s

Time:   15.00 s

Time:   20.00 s

Time:   25.00 s

54

# CASEDEF-MOTION

•*Motion07*: sinusoidal movement (**<mvrectsinu />, <mvrotsinu />, <mvcirsinu />**)

```xml
- <motion>
  - <objreal ref="4">
      <begin mov="1" start="0"/>
    - <mvrotsinu id="1" duration="5" next="2">
        <axisp1 x="0" y="0" z="2.85"/>
        <axisp2 x="0" y="1" z="2.85"/>
        <freq v="0.2"/>
        <ampl v="60"/>
        <_phase v="0"/>
      </mvrotsinu>
    - <mvrotsinu id="2" duration="5" next="1">
        <axisp1 x="0" y="0" z="2.85"/>
        <axisp2 x="0" y="1" z="2.85"/>
        <freq v="0.4"/>
        <ampl v="75"/>
      </mvrotsinu>
    </objreal>
  - <objreal ref="5">
      <begin mov="1" start="0"/>
    - <mvcirsinu id="1" duration="5" next="2">
        <ref x="0" y="-0.7" z="0.2"/>
        <axisp1 x="0" y="0" z="2.85"/>
        <axisp2 x="0" y="1" z="2.85"/>
        <freq v="0.2"/>
        <ampl v="60"/>
        <phase v="0"/>
      </mvcirsinu>
```
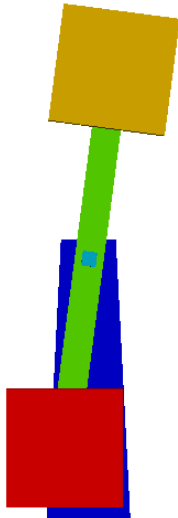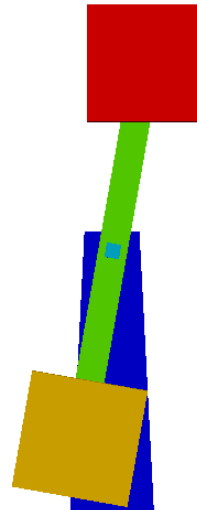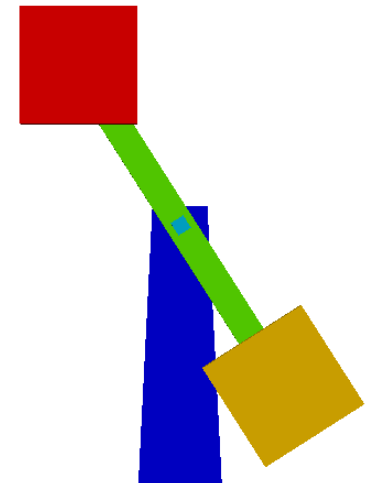
```xml
    - <mvcirsinu id="2" duration="5" next="1">
        <ref x="0" y="-0.7" z="0.2"/>
        <axisp1 x="0" y="0" z="2.85"/>
        <axisp2 x="0" y="1" z="2.85"/>
        <freq v="0.4"/>
        <ampl v="75"/>
        <phase v="0"/>
      </mvcirsinu>
    </objreal>
  - <objreal ref="6">
      <begin mov="1" start="0"/>
    - <mvrectsinu id="1" duration="5" next="2">
        <freq x="0.2" y="0" z="0"/>
        <ampl x="2.30" y="0" z="0"/>
        <phase x="0" y="0" z="0"/>
      </mvrectsinu>
    - <mvrectsinu id="2" duration="5" next="1">
        <freq x="0.4" y="0" z="0"/>
        <ampl x="2.55" y="0" z="0"/>
        <phase x="0" y="0" z="0"/>
      </mvrectsinu>
    </objreal>
  </motion>
```
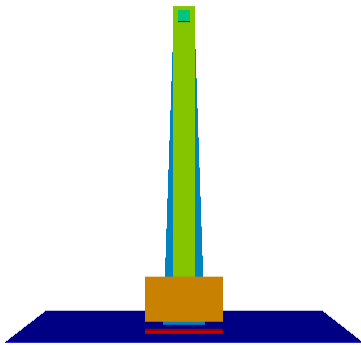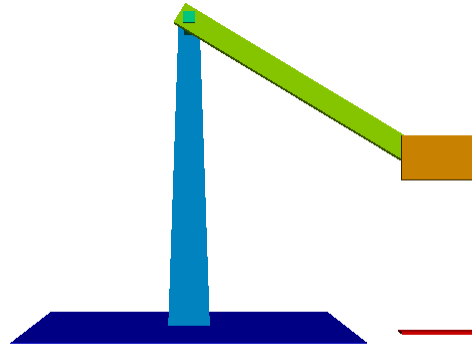
**<mvrectsinu>:** sinusoidal rectilinear movement

**<mvrotsinu>:** sinusoidal rotational movement

**<mvcirsinu>:** sinusoidal circular movement

**axisp1** first point of the rotation axis
**axisp2** second point of the axis
**freq** frequency
**ampl** amplitude
**phase** phase
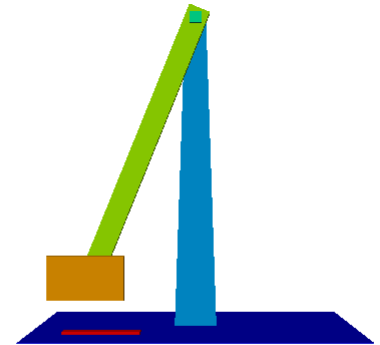
•*Motion07*: sinusoidal movement (**<mvrectsinu />, <mvrotsinu />, <mvcirsinu />**)
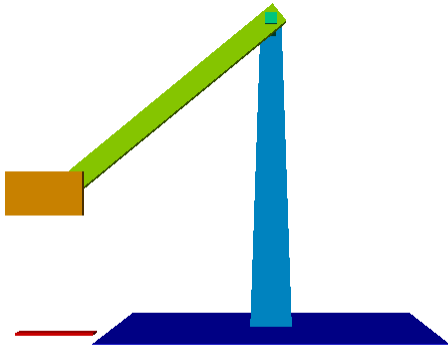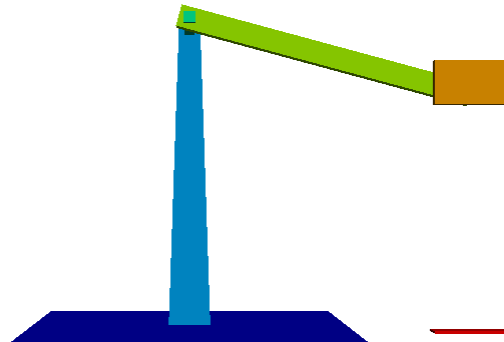


Time:   0.00 s

Time:   1.40 s

Time:   2.80 s

Time:   4.20 s

Time:   5.60 s

Time:   7.00 s

# CASEDEF-MOTION

•*Motion08*: predefined movement with data from an external file (**<mvpredef />**)

```xml
- <motion>
  - <objreal ref="200">
      <begin mov="1" start="0"/>
    - <mvpredef id="1" duration="10">
        <file name="motion08mov_f3.out" fields="4" fieldtime="0" fieldx="1" fieldy="2" fieldz="3"/>
      </mvpredef>
    </objreal>
  - <objreal ref="150">
      <begin mov="1" start="0"/>
    - <mvpredef id="1" duration="8" next="2">
        <file name="motion08mov_f3.out" fields="4" fieldtime="0" fieldx="1" fieldy="2"/>
      </mvpredef>
    - <mvrect id="2" duration="-1">
        <vel x="0" y="0" z="-0.02"/>
      </mvrect>
    </objreal>
  - <objreal ref="151">
      <begin mov="1" start="0"/>
    - <mvpredef id="1" duration="10">
        <file name="motion08mov_f3.out" fields="4" fieldtime="0" fieldx="1" fieldz="3"/>
      </mvpredef>
    </objreal>
  - <objreal ref="152">
      <begin mov="1" start="0"/>
    - <mvpredef id="1" duration="10">
        <file name="motion08mov_f3.out" fields="4" fieldtime="0" fieldy="2" fieldz="3"/>
      </mvpredef>
    </objreal>
</motion>
```
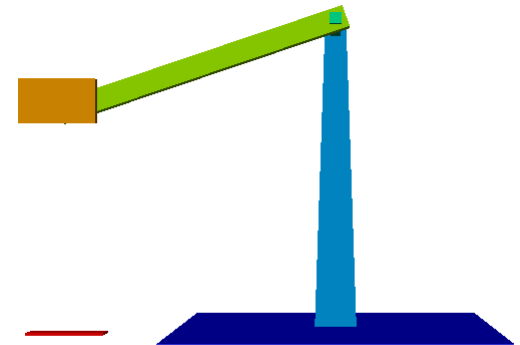
**<mvpredef>:** prescribed motion loaded from a file

**name** name of the file
**fields** number of columns of the file
**fieldtime** column with time
**fieldx** column with X-position
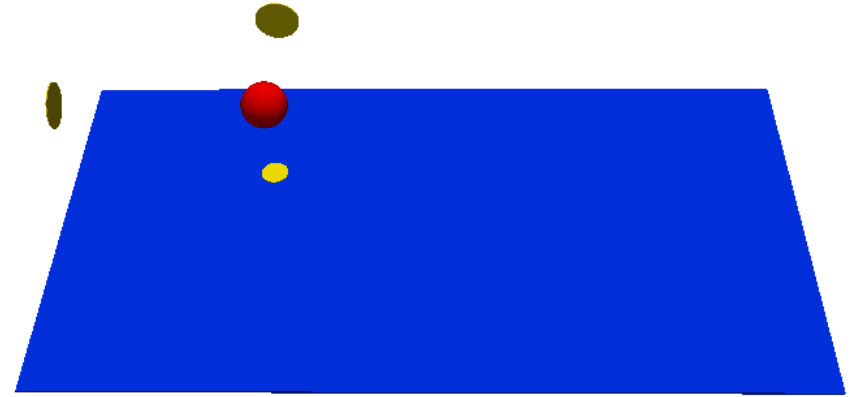**fieldy** column with Y-position
**filedz** column with Z-position

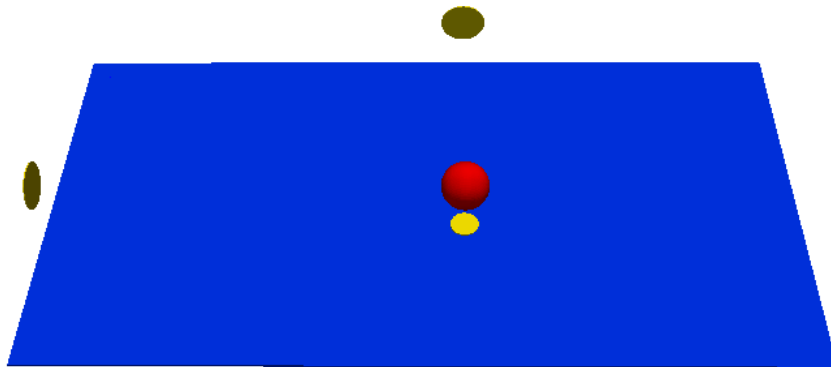# CASEDEF-MOTION

•*Motion08*: predefined movement with data from an external file (**<mvpredef />**)



Time: 0.00 s

Time: 3.00 s

Time: 6.00 s

Time: 9.00 s

# EXECUTION-PARAMETERS

```
- <execution>
  - <parameters>
      <parameter key="StepAlgorithm" value="1" comment="Step Algorithm 1:Verlet, 2:Symplectic (def=1)"/>
      <parameter key="VerletSteps" value="40" comment="Verlet only: Number of steps to apply Eulerian equations (def=40)"/>
      <parameter key="Kernel" value="1" comment="Interaction Kernel 1:Cubic Spline, 2:Wendland (def=1)"/>
      <parameter key="KernelGradientCorr" value="0" comment="Apply Kernel Gradient Correction (yes=1)"/>
      <parameter key="ViscoTreatment" value="1" comment="Viscosity Formulation 1:Artificial, 2:Laminar+SPS (def=1)"/>
      <parameter key="Visco" value="0.25" comment="Viscosity value"/>
      <parameter key="ShepardSteps" value="0" comment="Number of steps to apply Shepard density filter, 0=non applied (def=0)"/>
      <parameter key="DBCSteps" value="1" comment="Number of steps to update the density of the boundaries, 1 no correction (def=1)"/>
      <parameter key="DtIni" value="0.0001" comment="Initial time step"/>
      <parameter key="DtMin" value="0.00001" comment="Minimum time step (def=0.00001)"/>
      <parameter key="TimeMax" value="1.5" comment="Time of simulation"/>
      <parameter key="TimeOut" value="0.015" comment="Time between output files"/>
      <parameter key="IncZ" value="2" comment="Increase of Z+"/>
      <parameter key="PartsOutMax" value="1" comment="Allowed percentage of fluid particles out the domain (def=1)"/>
  </parameters>
</execution>
```

**<parameters>:**

Consits on a set of parameters for the DualSPHysics execution

*StepAlgorithm, Kernel, ViscoTreatment, Shepard, DBC* are based on SPHysics formulation

GenCase keeps this information in the new XML to be loaded by DualSPHysics