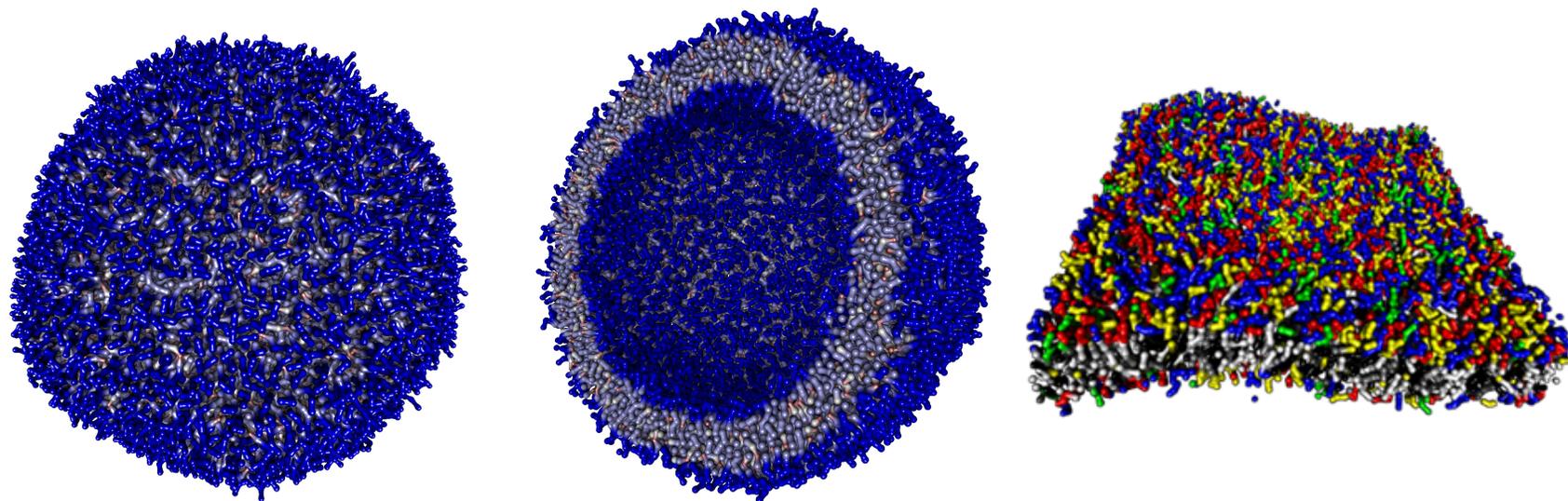


UnityMol WebGL: Martini Coarse-Grain systems visualization



Xavier Martinez
CNRS-LIMSI Equipe VENISE, Orsay

Sébastien Doutreligne
CNRS-UPR 9080 Laboratoire de Biochimie
Théorique, Paris

Matthieu Chavent
Institut de Pharmacologie et Biologie
Structurale, Toulouse

Marc Baaden
CNRS-UPR 9080 Laboratoire de Biochimie
Théorique, Paris



UnityMol:

- Based on a game engine
- Framework to quickly develop :
 - GUI with Unity UI system
 - Molecular visualizations
 - Interaction metaphores

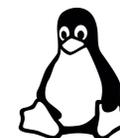
Windows

Tested on 7/8/8.1/10



Linux

Tested on Ubuntu 12.04 ->16.04



macOS

Tested on 10.6 -> Sierra



Android

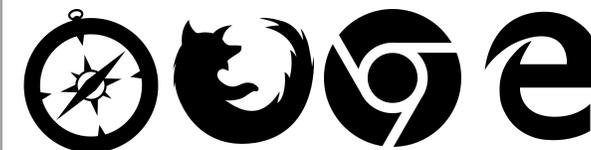
Beta tested on 6.0/7.0



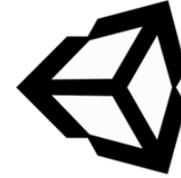
WebGL



Safari / Firefox / Chrome / Edge



Unity3D



- Cross-platform Game Engine
- All purpose: 2D/3D games, simulations with Nvidia PhysX , **scientific visualization**
- Mostly free + widely adopted = good support
- C# / Javascript (UnityScript) / Boo

VR & AR
All platforms supported



Android VR
Daydream & Cardboard



Gear VR
Samsung VR



HTC Vive & Oculus Rift
Desktop



Playstation VR
Console

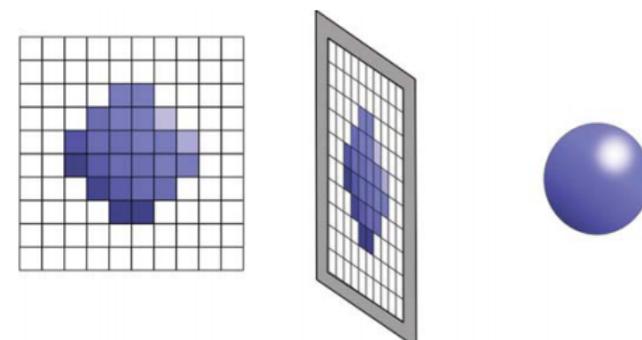
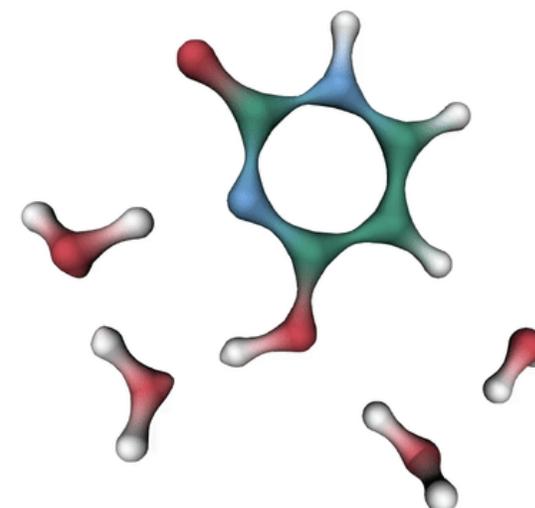
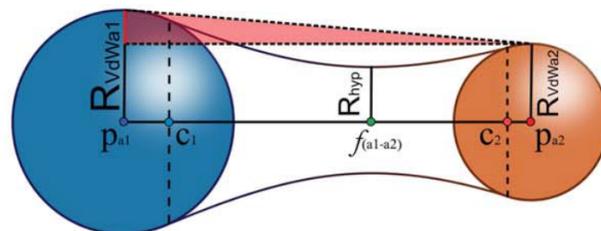


Microsoft HoloLens
AR



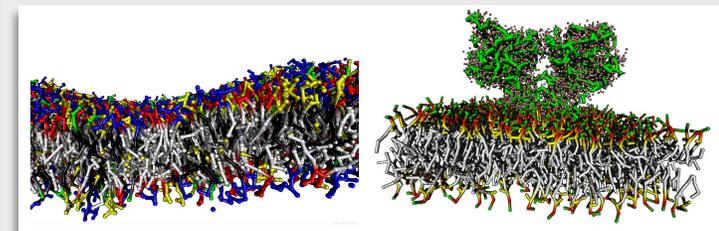
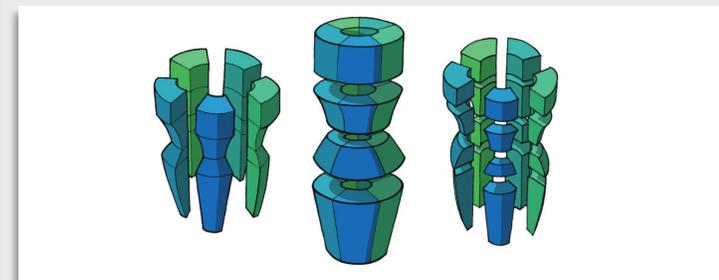
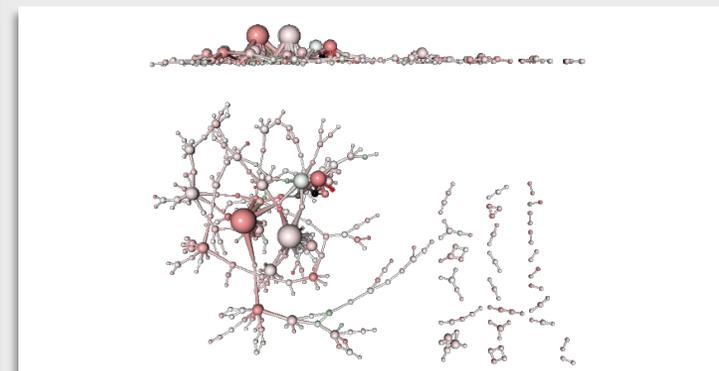
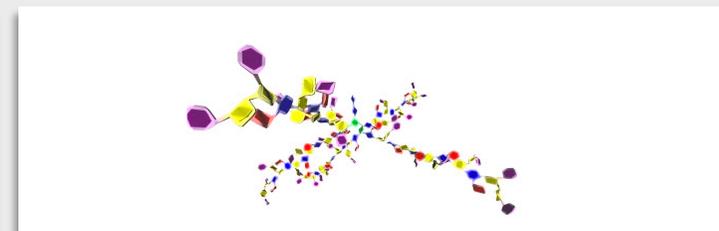
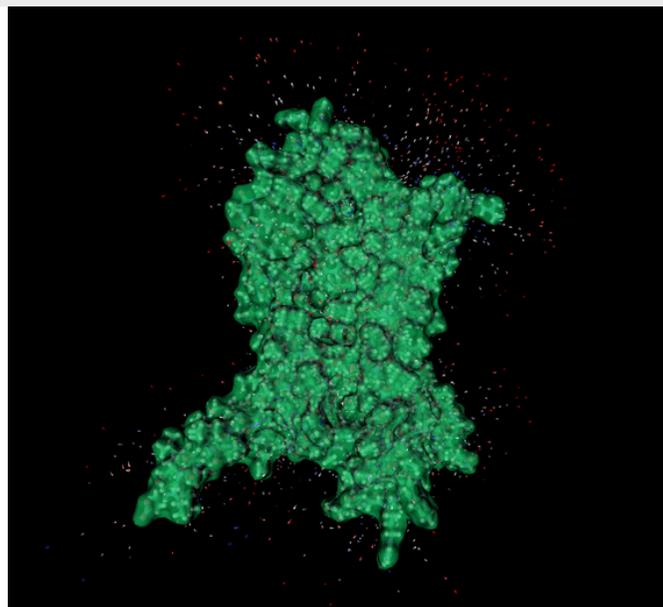
Hyperballs

- Sphere raycasting
- Computed on the GPU via a shader: dynamic
- High performance
- One shader = all representations:
 - VDW
 - Balls&Sticks
 - Licorice



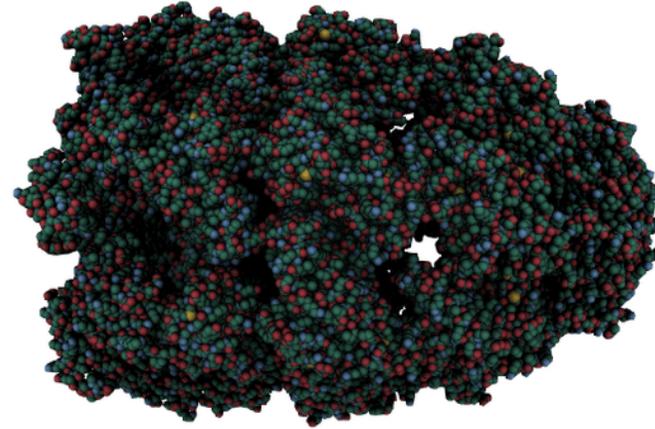
Features

- Protein/Sugar/DNA/RNA/CG visualization in \neq representations
- Biological Network visualization
- QM & MD trajectories visualization
- Interactive Molecular Dynamics (IMD)
- Field Lines, navigation guidée, modélisation sémantique...



Recent optimizations

- Code refactoring
- Overall performance gain
- 500k atoms on a laptop @ 90+fps



New possibilities
FPS boost

Large systems
All-atoms & Coarse Grain

Trajectories
Gromacs & AMBER

IMD
Gromacs

VR !
All of the above in VR

Unity binding with C/C++ libraries:

- Native C/C++ code:
 - Performance
 - Existing libraries
- C# -> C bindings



A lot more...

Example: Trajectories

- XTC reader using Gromacs C library

```
namespace Trajectories {
    public enum XDRStatus
    { exdrOK, exdrHEADER, exdrSTRING, exdrDOUBLE,
      exdrINT, exdrFLOAT, exdrUINT, exdr3DX, exdrCLOSE, exdrMAGIC,
      exdrNOMEM, exdrENDOFFILE, exdrFILENOTFOUND, exdrNR };

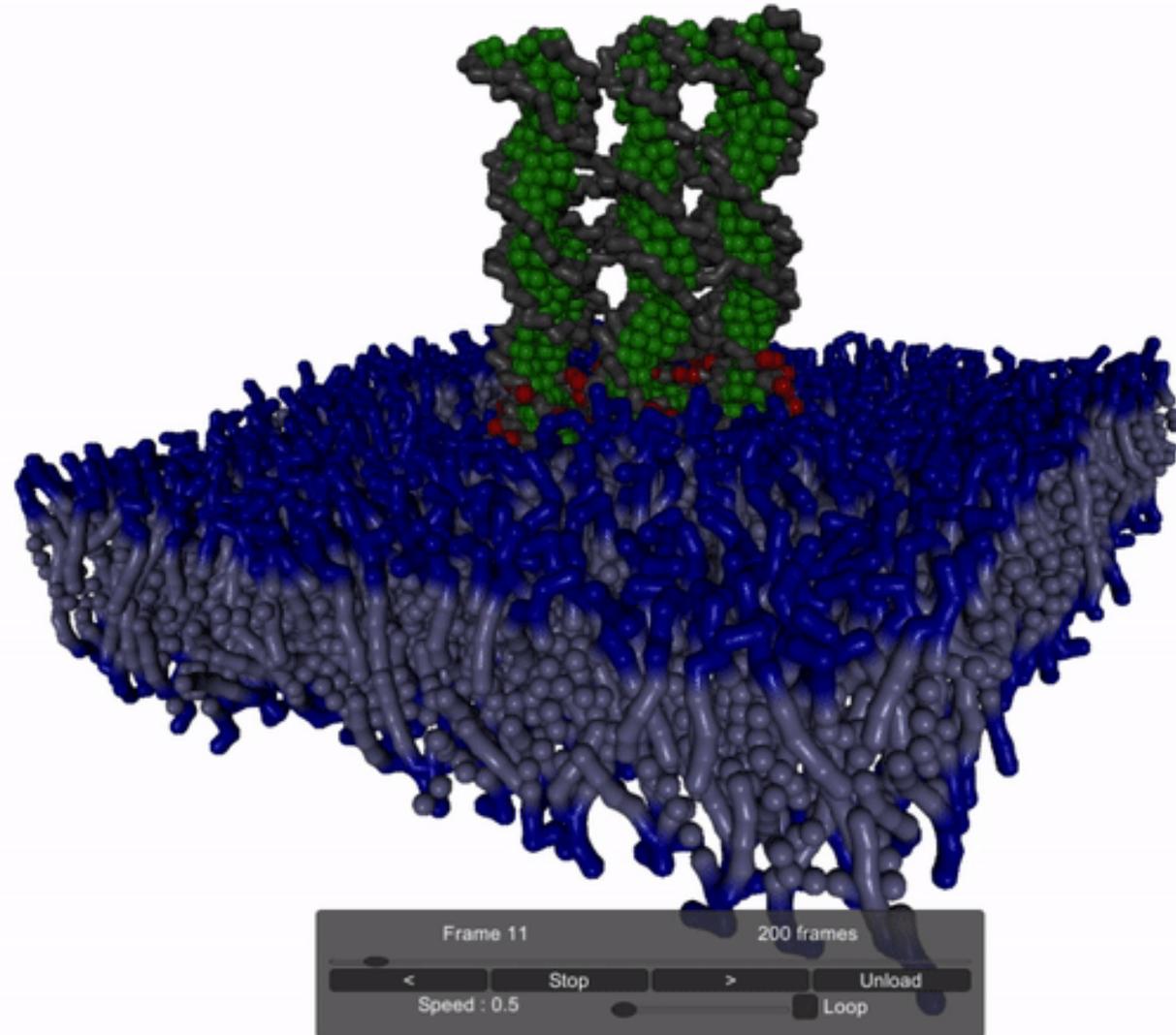
    // Provides function prototypes to use the xdrfile library.
    public class XDRFileWrapper {
        // Opens a trajectory file located at "path" using the provided mode.
        // mode = "r" for read, mode = "w" for write.
        // Returns a file pointer to an xdr file datatype, or NULL if an error occurs.
        [DllImport ("xdrfile", CallingConvention=CallingConvention.Cdecl)]
        public static extern System.IntPtr xdrfile_open([In] string path, [In] string mode);

        // Closes a previously opened trajectory file passed in argument.
        // Returns 0 on success (XDRStatus.exdrOK), non-zero on error.
        [DllImport ("xdrfile", CallingConvention=CallingConvention.Cdecl)]
        public static extern XDRStatus xdrfile_close([In] System.IntPtr xfp);

        // Returns the number of atoms in the xtc file into *natoms.
        // Returns 0 on success (XDRStatus.exdrOK), non-zero on error.
        [DllImport ("xdrfile", CallingConvention=CallingConvention.Cdecl)]
        public static extern XDRStatus read_xtc_natoms([In] string filename, ref int natoms);

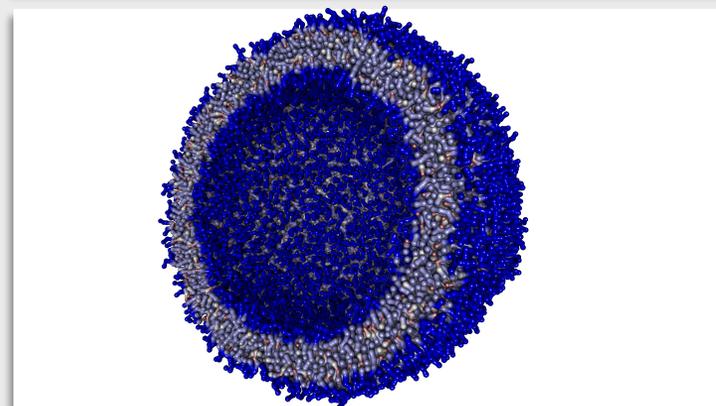
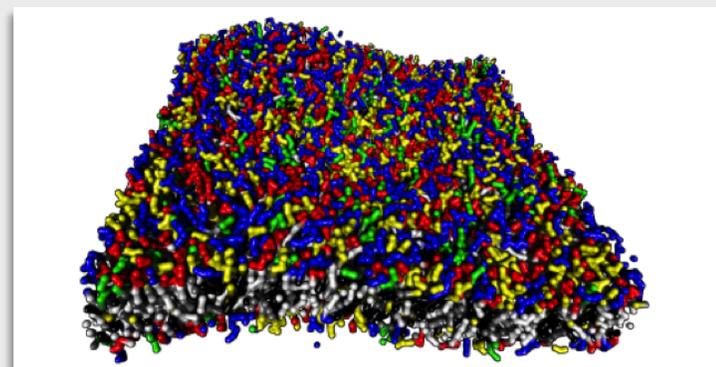
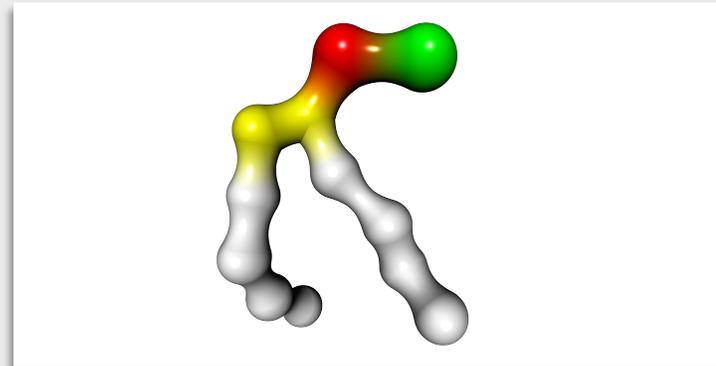
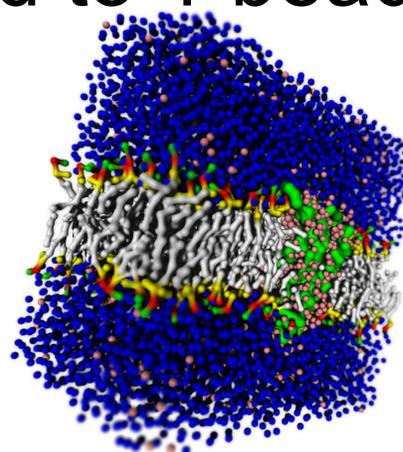
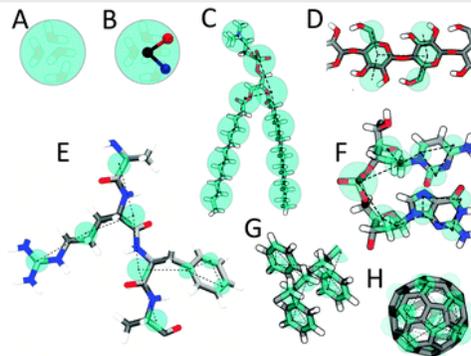
        // Reads one frame of an opened xtc file.
        // Returns 0 on success (XDRStatus.exdrOK), non-zero on error.
        [DllImport ("xdrfile", CallingConvention=CallingConvention.Cdecl)]
        public static extern XDRStatus read_xtc(System.IntPtr xd, int natoms, ref int step, \
            ref float time, float[,] box, [In, Out] float[] x, ref float prec);
    }
}
```

IFB BioStruct: Workshop Web Viewers



Martini coarse grain

- Coarse grain: 4 atoms mapped to 1 bead
- 4 types of interaction
 - Polar
 - Non-polar
 - Apolar
 - Charged
- Gromacs/Gromos/NAMD
- Larger systems, longer MD simulations



File parsing

- Color / Radius file
- ITP file: topology
- PDB/GRO file

```
#Residue/Lipid CGname colorR colorG colorB radius
```

```
DPPC NC3 0.9 0.0 0.6 0.5  
DPPC P04 0.8 0.0 0.6 0.5  
DPPC GL1 0.7 0.0 0.6 0.5  
DPPC GL2 0.6 0.0 0.6 0.5  
DPPC C1A 0.5 0.0 0.6 0.5  
DPPC C2A 0.4 0.0 0.6 0.5  
DPPC C3A 0.3 0.0 0.6 0.5  
DPPC C4A 0.2 0.0 0.6 0.5  
DPPC C1B 0.1 0.0 0.6 0.5  
DPPC C2B 0.0 0.0 0.4 0.5  
DPPC C3B 0.0 0.0 0.2 0.5  
DPPC C4B 0.0 0.0 0.0 0.5
```

```
[moleculetype]  
; molname nrexcl  
DPPC 1  
  
[atoms]  
; id type resnr residu atom cgnr charge  
1 Q0 1 DPPC NC3 1 1.0  
2 Qa 1 DPPC P04 2 -1.0  
3 Na 1 DPPC GL1 3 0  
4 Na 1 DPPC GL2 4 0  
5 C1 1 DPPC C1A 5 0  
6 C1 1 DPPC C2A 6 0  
7 C1 1 DPPC C3A 7 0  
8 C1 1 DPPC C4A 8 0  
9 C1 1 DPPC C1B 9 0  
10 C1 1 DPPC C2B 10 0  
11 C1 1 DPPC C3B 11 0  
12 C1 1 DPPC C4B 12 0  
  
[bonds]  
; i j funct length force.c.  
1 2 1 0.47 1250  
2 3 1 0.47 1250  
3 4 1 0.37 1250  
3 5 1 0.47 1250  
5 6 1 0.47 1250  
6 7 1 0.47 1250  
7 8 1 0.47 1250  
4 9 1 0.47 1250  
9 10 1 0.47 1250
```

Web integration

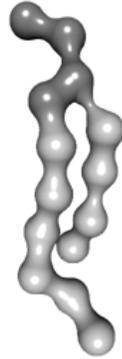
- **UnityLoader.js**
 - Web ->Unity game engine
 - SendMessage(GameObject, Function, Argument)

```
public void loadCGFiles(string args){  
    string delim = "$$$";  
    string[] allargs = args.Split(new string[] {delim}, System.StringSplitOptions.RemoveEmptyEntries);  
    groFile = allargs[0];  
    StartCoroutine(loadGROFile(float.Parse(allargs[1]),float.Parse(allargs[2])));  
}
```

```
<script src="Release/UnityLoader.js"></script>  
  
<script>  
function changelipid(lipid_name,shrink,camdist) {  
    nameGro = lipid_name + '.gro';  
    args = nameGro+"$$$"+shrink.toString()+"$$$"+camdist.toString();  
    SendMessage('loadCG', 'loadCGFiles',args);  
    document.getElementById('myspan').textContent=lipid_name;  
}  
</script>  
  
<input id="ChangeLipid" type="image" src="images/BNSM_box.png" onclick="changeLipid('BNSM','0.17','20');" />  
</script>
```

Result

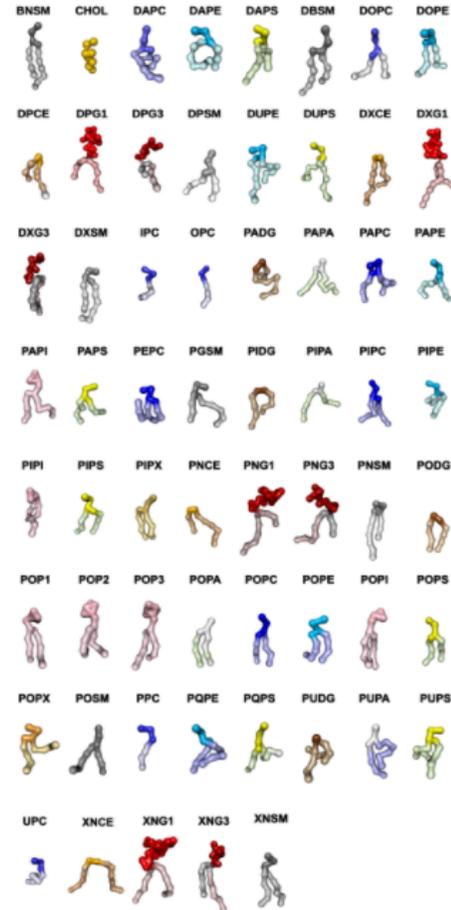
BNSM



UNITYMOL CG

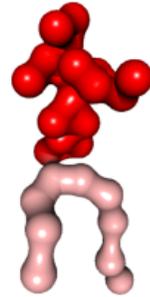
Plasma Membrane Lipids

The lipid structures were extracted from a [complex membrane model](#) developed by the [Memiris lab](#) at the University of Göttingen. Here you can see the diversity of lipids in this membrane. The type of force field used is [MARTINI](#), a coarse grain forcefield where one particle depicts 3-4 heavy atoms. We used [unitymol](#) to render these lipid models. Please use Firefox, Chrome, or Safari browsers to display this page: unfortunately IE does not display unity WebGL windows.



Result

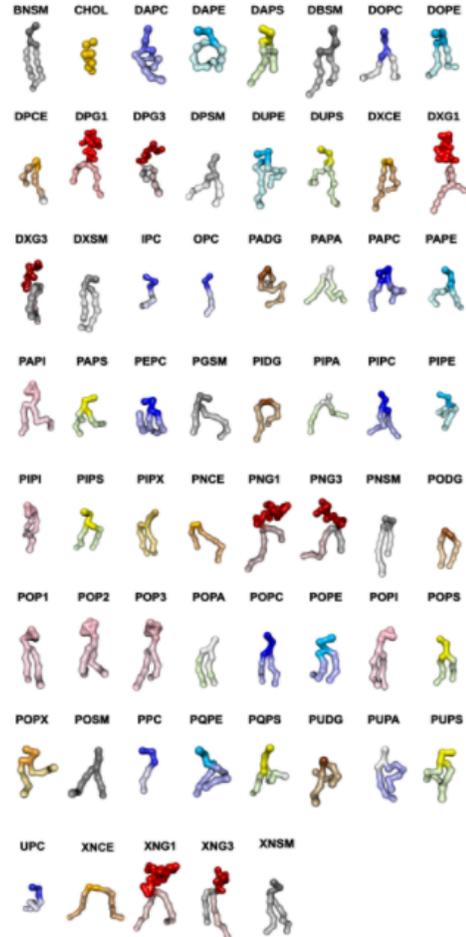
DPG1



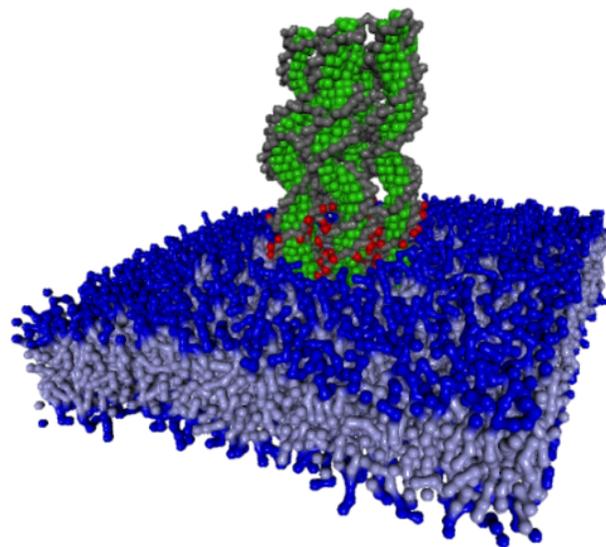
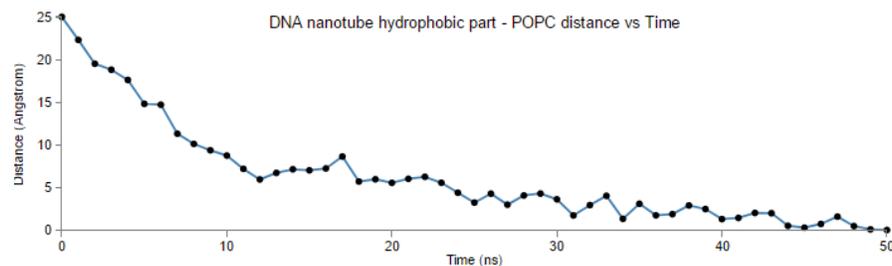
UNITYMOL CG

Plasma Membrane Lipids

The lipid structures were extracted from a [complex membrane model](#) developed by the [Martinko lab](#) at the University of Groningen. Here you can see the diversity of lipids in this membrane. The type of force field used is [MARTINI](#), a coarse grain forcefield where one particle depicts 3-4 heavy atoms. We used [unitymol](#) to render these lipids models. Please use Firefox, Chrome, or Safari browsers to display this page: unfortunately IE does not display unity WebGL windows.



Result



UNITYMOL CG

DNA Nanotube

Interactive Graph combined with UnityMol CG

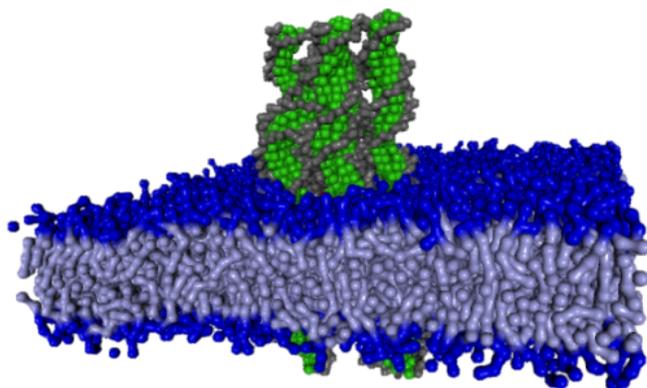
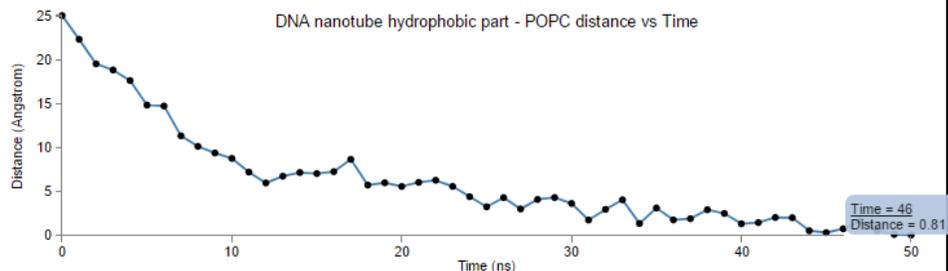
Combining UnityMol WebGL viewer with other tools such as [D3.js](#) helps to design new ways to explore the data. It is now possible to create interactive figures. An example is shown here for a DNA nanotube-membrane complex. Here you can pass the mouse on top of each point to display nanotube with respect to membrane at specific time step.

You have to wait a bit to see the first scene depicting the DNA nanotube model (in green) with its hydrophobic decorations (red) inserted in a POPC bilayer (in blue). When you pass the mouse on top of a point a little blue box appears. In this box itself, you can then click on the time to change the scene and display the position of the DNA nanotube in the membrane. Please wait a bit as the scene is quite big and can create some delay to display.

More information about this model can be found in the recent [article](#) published in Nature Communications. We would like to thank [Vishal Mainj](#) for providing the data.



Result



UNITYMOL CG

DNA Nanotube

Interactive Graph combined with UnityMol CG

Combining UnityMol WebGL viewer with other tools such as [D3.js](#) helps to design new ways to explore the data. It is now possible to create interactive figures. An example is shown here for a DNA nanotube-membrane complex. Here you can pass the mouse on top of each point to display nanotube with respect to membrane at specific time step.

You have to wait a bit to see the first scene depicting the DNA nanotube model (in green) with its hydrophobic decorations (red) inserted in a POPC bilayer (in blue). When you pass the mouse on top of a point a little blue box appears. In this box itself, you can then click on the time to change the scene and display the position of the DNA nanotube in the membrane. Please wait a bit as the scene is quite big and can create some delay to display.

More information about this model can be found in the recent [article](#) published in Nature Communications. We would like to thank [Vishal Maingi](#) for providing the data.



URLs:

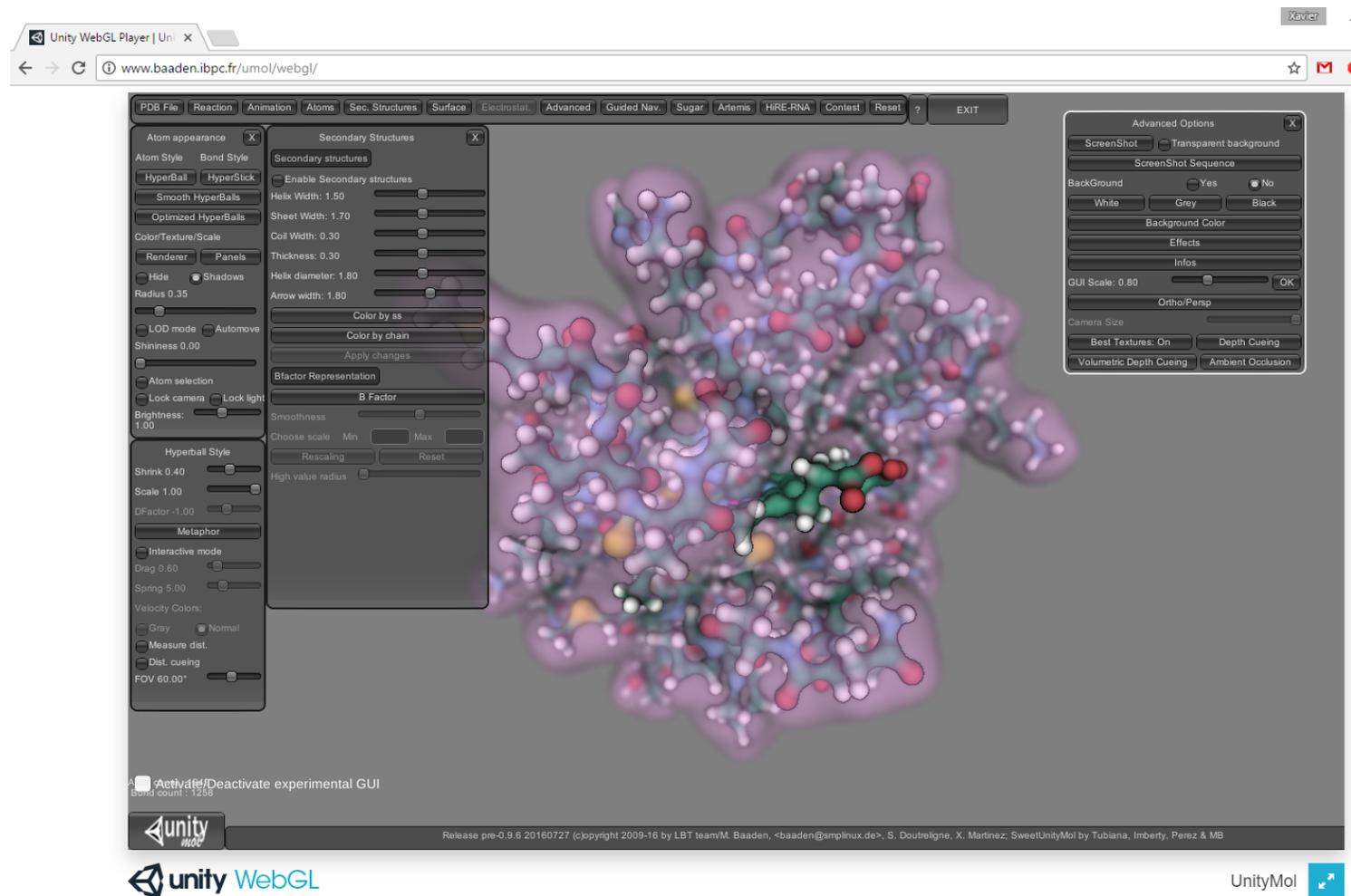
<http://matthieuchavent.com/PM/PM.html>

<http://matthieuchavent.com/PM/DNA.html>

UnityMol WebGL

- Initial test:

<http://v.ht/uMol>





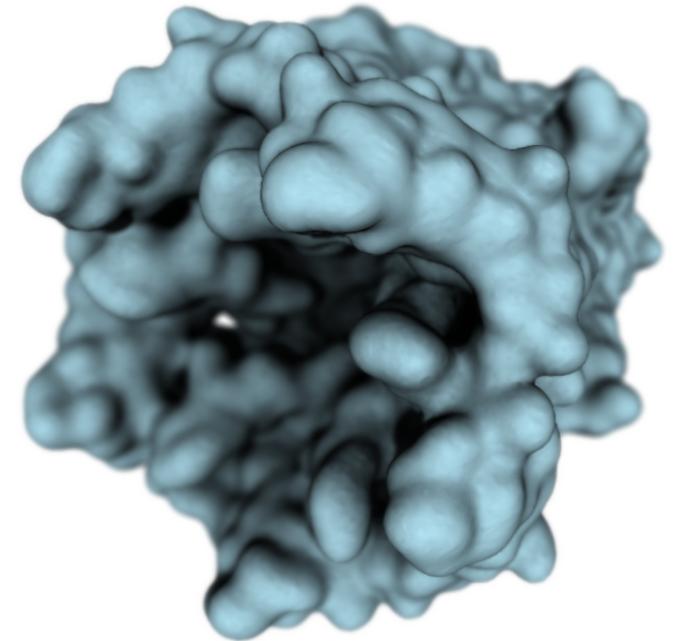
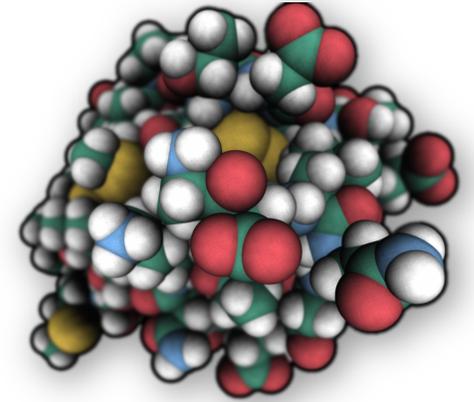
Work in Progress (WebGL)

- Memory management
- Better file input
- Read trajectories in WebGL builds
- Unity still working on WebGL (2.0)
- Unity has to fix WebGL on mobiles

Work in Progress (UnityMol)



- VR & AR: **WebVR**
- New effects: ambient occlusion for atoms & surfaces
- Surface & Ribbons generation in C/C++ library
- New & clean future proof implementation

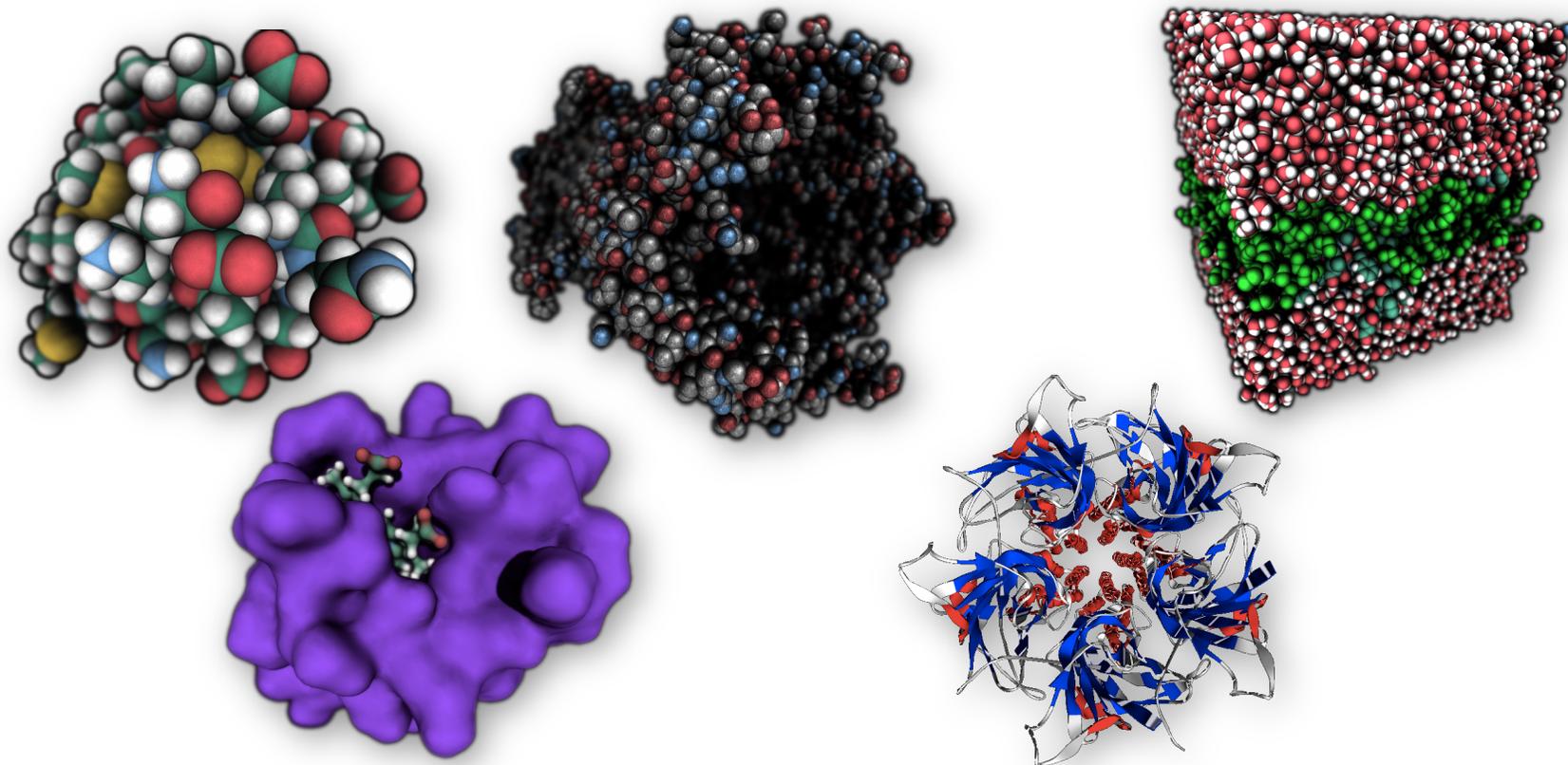


Conclusion:

- Lots of possibilities: nearly all game engine/
UnityMol features available in WebGL
- Hyperball shader performance can still increase
- UnityMol framework can grow with libraries as
modules

<http://unitymol.sourceforge.net/>

Thank you for your attention



Xavier Martinez
CNRS-LIMSI Equipe VENISE, Orsay

Sébastien Doutreligne
CNRS-UPR 9080 Laboratoire de Biochimie
Théorique, Paris

Matthieu Chavent
Institut de Pharmacologie et Biologie
Structurale, Toulouse

Marc Baaden
CNRS-UPR 9080 Laboratoire de Biochimie
Théorique, Paris

