

Autodesk Maya modeling, animation, scripting and C++ programming 2017-18

lionel.reveret@inria.fr

Cours ENSIMAG, Ingénierie de l'Animation 3D

Goals

- Discover a professional tool for 3D production
 - Practical implementation of theoretical concept
 - Gain experience on a software that is a reference in the digital media industry
- Learn the role of programmers in 3D workflows
 - To cooperate with artists and engine programmers
 - Developing tools
 - Scripts (MEL / Python)
 - Plug-ins as dynamic library (C++ API / Python API)

Organization & Evaluation

- Introduction to Maya (6h)
- Development project (9h)
- Evaluation (3h)
 - Project presentation

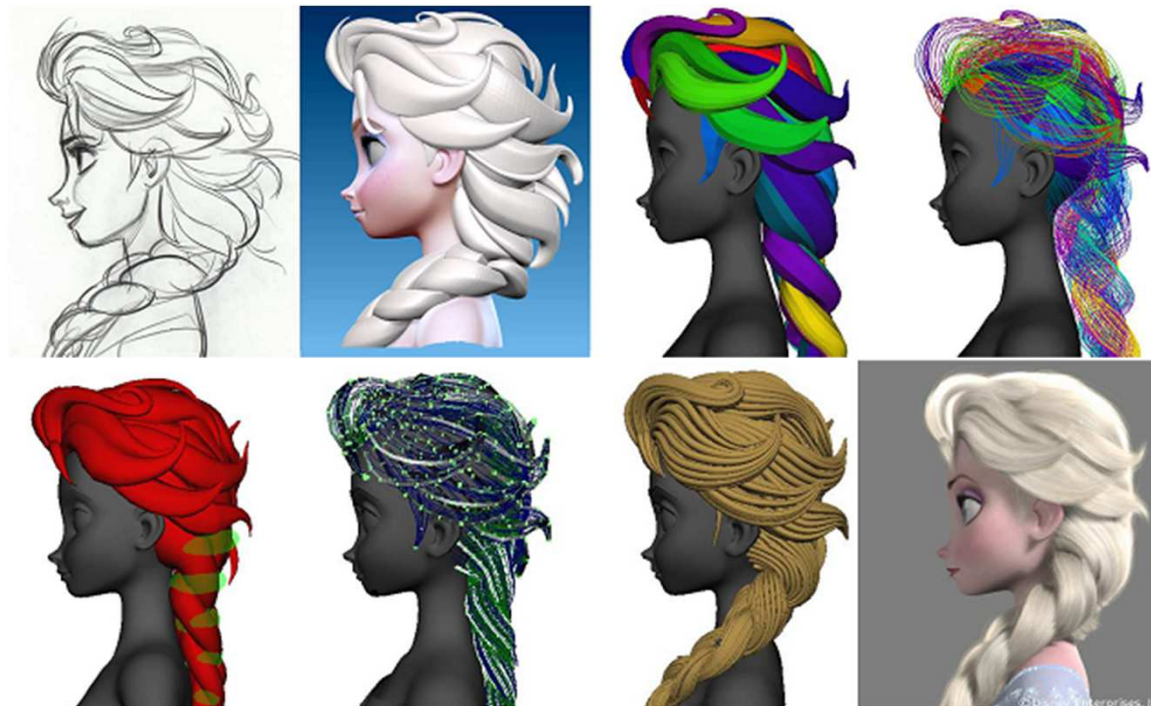
3D Programming

- Different software categories
 - Libraries
 - Low level: OpenGL, DirectX, CUDA, OpenCL
 - Higher level: Qt, OpenInventor, etc
 - Engines
 - Rendering, Animation, Physics, All-in-one (Unreal, CryEngine, etc)
 - Artist software
 - General purpose: 3ds Max, Maya, Blender
 - Rendering: Mental Ray, Arnold, Renderman
 - Animation: MotionBuilder, Cinema4D
 - Modeling: Rhinoceros, ZBrush, Mudbox

3D Programming in the Digital Media Industry

- Animation Studios

- Maya and 3ds Max are the reference
- Proprietary suites are used in the biggest studios (Disney, DreamWorks, Pixar)
 - ex: Disney's Tonic tool for hair modeling & simulation

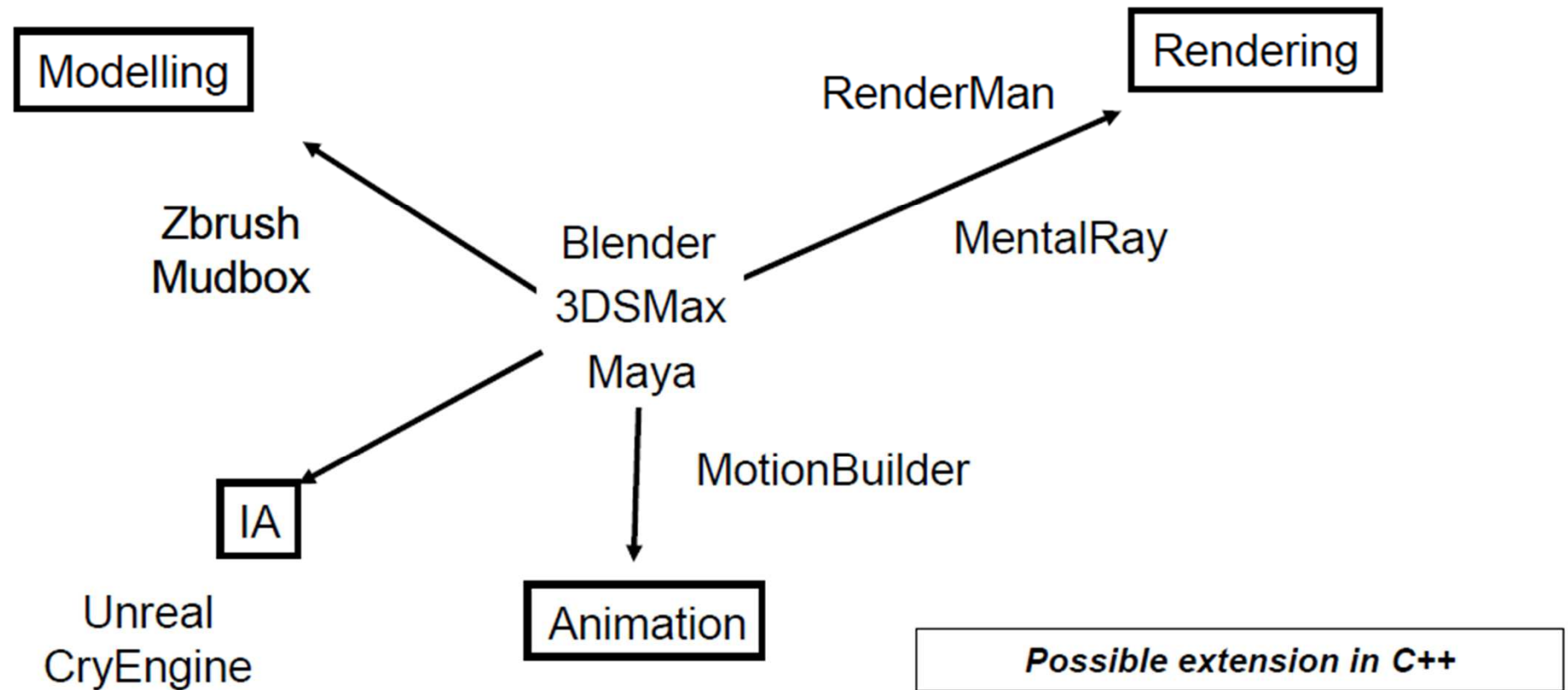


3D Programming in the Digital Media Industry

- Game Studios
 - Most studios use Maya or 3ds Max for 3D assets
 - Proprietary tools as well as file formats (to protect assets) for run-time
 - Assets are used by the 3D engine of an end-user software

3D Programming in the Digital Media Industry

- Extended workflow



Why Autodesk Maya ?

- Interactive application for:
 - Modeling
 - Polygons, NURBS, Subdivision, Texture placement
 - Animation
 - keyframe, skeleton, physics
 - Rendering
 - Interactive (real-time visualization), off-line (ray-tracing)
- Open software architecture (script and C++), not open source
- Multi-platform (formats, script and code)
- Nice Documentation support

Project summary

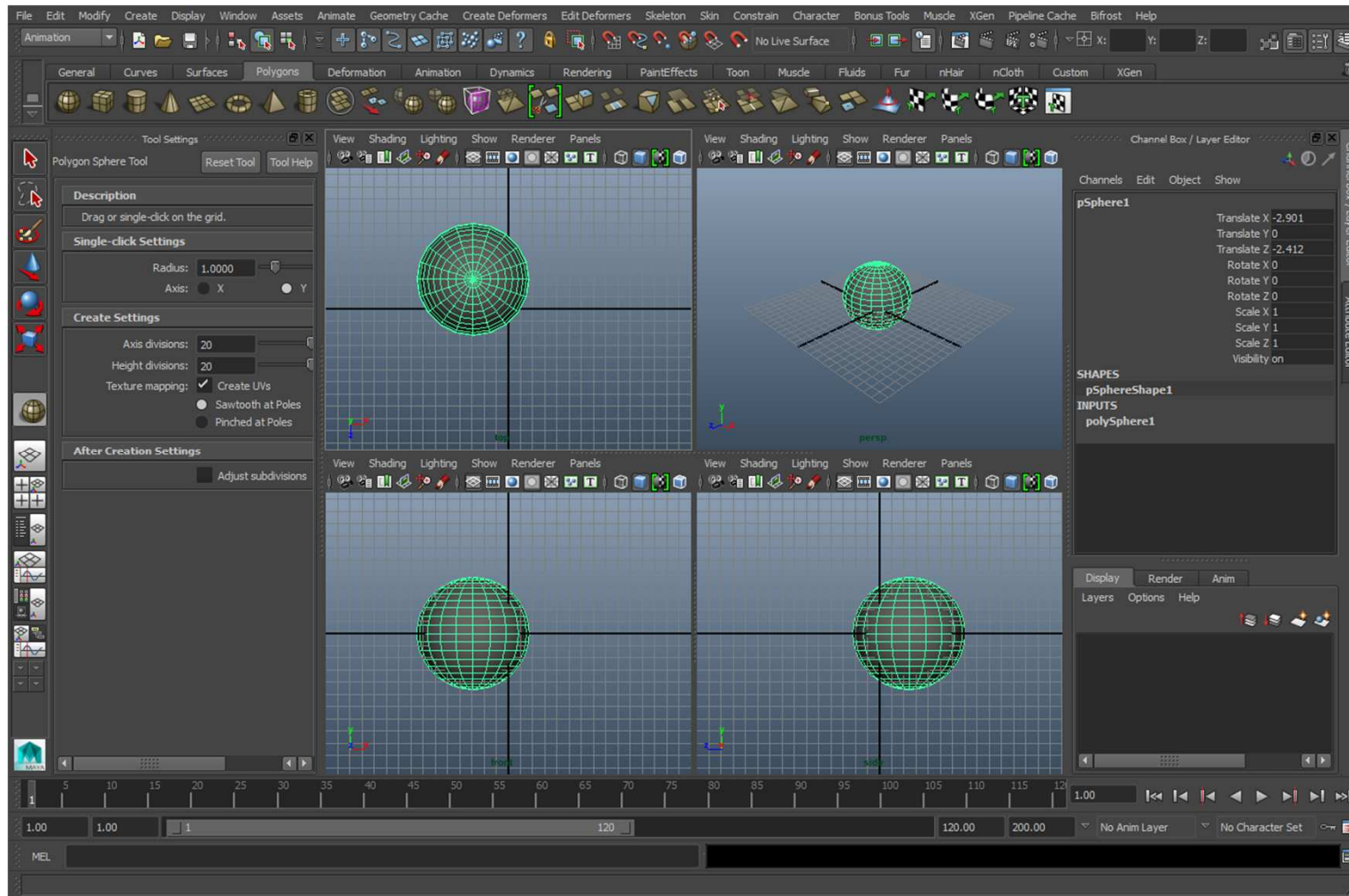
- Development of a tool related to animation
 - Specific deformation technique: Green Coordinates
- 4 languages will be addressed
 - Maya Embedded Language (MEL) Script
 - Maya Python Script
 - Python using Maya API
 - C++ using Maya's C++ API
 - Integrating the algorithm into Maya's core mechanics

But first !

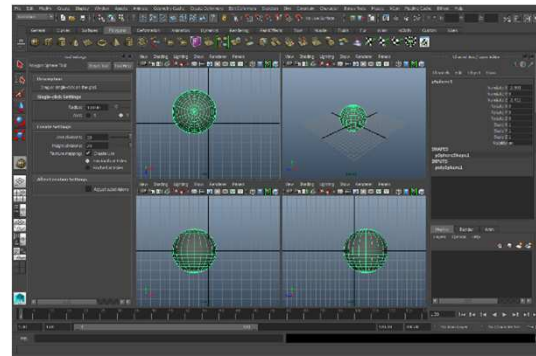
You will discover and manipulate this software a bit

(Artists have around 6 months of full-time training to use it properly,
so don't expect to be experts at the end of this course after 18 hours only...)

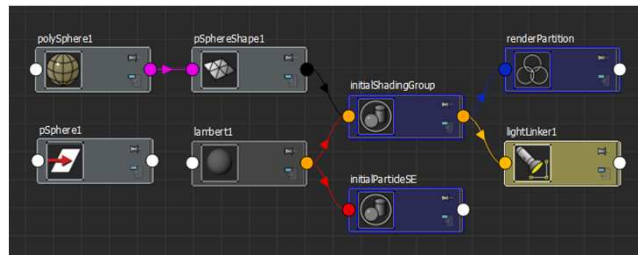
Maya Interface



Maya's software architecture



Interface



DG (Nodes & Attributes)
DAG (Scene Graph)

```
MEL Python
1 updateRendererUI;
2 CreatePolygonSphere;
3 setToolTo CreatePolySphereCtx;
4 polySphere -ch on -o on -r 4.602758 ;|
5 select -cl ;
6 select -r pSphere1 ;
```

MEL / Python

```
MStatus status;
MDagPath dag_path;
MFnIKJoint fn_joint;

// MITDag::kDepthFirst used to assure hierarchical order :)
MITDag it_dag(MITDag::kDepthFirst, MFn::kJoint, &status);
if (status != MStatus::kSuccess)
    FAILURE("SkIWriter: MITDag::MITDag()");

int num_joints = 0;
for (; !it_dag.isDone(); it_dag.next())
{
    SklBone bone;
    num_joints++;
    it_dag.getPath(dag_path);
    data_.joints.append(dag_path);
    fn_joint.setObject(dag_path);
    MString jointName = fn_joint.name();
    strcpy_s(bone.name, SklBone::kNameLen, jointName.asChar());
    MQuaternion rotation, axe;
    axe = fn_joint.rotateOrientation(MSpace::kTransform);
    fn_joint.getRotation(rotation, MSpace::kWorld); // since it's kWorld it
```

C++ API

MEL (Maya Embedded Language) / Python module

- create/edit objects, query/set attributes
 - geometry, animation, computation nodes
- Algorithm
 - control flow, data structures, procedures
- GUI (Maya own programming widgets)
 - input handling

Maya API C++

- Programmable nodes
 - input: geometry[], float[], time, etc
 - output: geometry[], float[], color[], etc
 - Exemples : shaders, skinning
- Custom File I/O
- Custom tools (3D HCI widgets)
 - access to OpenGL context of interactive view
- Stand-alone application
 - use Maya API but no need for Maya Interface
 - open Maya binary files

Maya Learning tools

- Several links in Maya Help Menu
- Additional tutorials and content from Autodesk website

Maya and 3D on the web

- <https://www.autodesk.com/education/free-software/all>
 - FREE version for **student**
 - Go for education community account
- www.highend3d.com
 - great source of scripts and plug-ins (mostly free), for Maya and others
- www.turbosquid.com
 - A lot of models (some free ones)