

MEL

in general, use Help->Tutorials, Help->Maya Help
for syntax, use Help->Maya Help; search "MEL" and find 'MEL for
programmers"
will cover various ways to use MEL to model and animate
search web for 'Maya MEL scripts' - many resources on-line; e.g.,
highend3d.com/maya/downloads/mel_scripts/animation

=====
MEL concepts

UI -> MEL engine -> Dependency Graph (DG)
everything is represented as nodes, attributes, connections
for everything that Maya does, it uses MEL
MEL interacts with the DG
data flow, push-pull evaluation

The scene - hierarchical organization of scene components
top down hierarchy (acyclic)
in Maya: directed acyclic graph (DAG)

visualize the dependency graph using the hypergraph

nodes: input, output, compute(); input & output values are *local* to the
node
input/output distinction is not explicit - all just attributes
attribute: name & data type; simple (e.g. int) or complex (e.g. NURBS
surface)
dependent attributes are derived from, or computed from, other attributes

special time node named *time1*
playback controls modify outTime value of time1 node

node attributes can be connected together
node does not know about its connections
attribute can have multiple destinations

DAG nodes: directed acyclic graph
hierarchy: parent-child nodes
scene hierarchy
subset of DG nodes (input-output connections)
e.g. transform, shape nodes
create an object: transform node -> shape node
not a tree - a directed graph with no cycles

Parenting
transform hierarchy, e.g. human figure
Edit->Group ?

DAG paths
path from root node to leaf node
uses '|', e.g. | torse | upper arm | lower arm |

Parameter space

parameter space on the surface of an object (2D); uses -> for DAG path
parameter space along a curve (1D)

INSTANCES

transform node refers to shared shape node
multiple paths to shape node

Dependency graph updating
conceptually data flow
implemented with push-pull
only update values required to get requested value

=====

MEL command line:

sphere;

=====

Connections

Connection Editor: Window->General Editors->Connection Editor
select Object1, in C.E., reload left (from)
select Object2, in C.E., reload right (to)
select 'translate y' from left side and 'translate x' from right side
if AutoConnect is selected (indicated by italicizing attributes)
connection is made
else need 'Connect' button at bottom (select File->Auto Connect)
move Object1 up and down

make connection in MEL
connectAttr Object1.tx Object2.ty

=====

Driven Keys

You create driven keys in the Set Driven Key window.
To display the Set Driven Key window:
* From the Animation menu bar or Hotbox, choose Keys > Set Driven Key
> Set-.
If you select an object before making the menu selection, the Set
Driven Key window lists and selects it as the driven object by default.
or
* In the Attribute Editor or Channel Box, click the attribute to be
driven, then click the right mouse button over the attribute and select
Set Driven Key.
The object and attribute you clicked in the Attribute Editor or
Channel Box are listed and selected in the Set Driven Key window as the
driven object and attribute.

Linking two attribute values

You link a pair of attribute values from the Set Driven Key window as
follows:

- 1 Open the Set Driven Key window.

- 2 In the workspace or Outliner, select the object that has the attribute to be the driver.
- 3 In the Set Driven Key window, click Load Driver.
The object and its attributes appear in the upper part of the window.
- 4 In the Set Driven Key window, select the attribute to be the driver.
- 5 In the workspace or Outliner, select the object that has the attribute to be driven.
Note that the driven and driver attributes can be in the same object.
- 6 Click Load Driven.
The object and its attributes appear in the lower part of the window.
- 7 In the Set Driven Key window, select the attribute to be the driven attribute.
- 8 In the Channel Box, Attribute Editor, or elsewhere, set values for the driver and driven attributes.
- 9 Click Key or select Key > Set.
Maya creates a key that links the selected attributes at the current values. The object with the driven attribute becomes magenta in the workspace. The color indicates the object has a driven attribute.
- 10 Set new values for the driver and driven attributes.
- 11 Click Key or select Key > Set.
Maya creates a key that links the attributes at their new values.
You can link the attributes with as many keys as necessary. The default interpolation between the keys is linear.
- 12 Close the Set Driven Key window.
- 13 Try various values for the driver attribute to check that the driven attribute responds appropriately.
To tune the relationship between the driver attribute and driven attribute, select Window > Animation > Graph Editor to use the Graph Editor. You can cut, copy, paste, and delete driven keys with the Graph Editor.
The prior procedure used buttons in the Set Driven Key window. You can use menu selections instead of the buttons to get the same result: You can examine objects with their driven key attribute settings. This is helpful when you're editing the Graph Editor curve or setting additional driven keys.

To see objects with their driven key attribute values:

- 1 In the workspace or Outliner, select the object that has the driven attribute.
- 2 From the Set Driven Key window, select Key > Go To Next or Key > Go To Previous.
or
From the Animation menu bar, select Keys > Set Driven Key > Go To Previous or Keys > Set Driven Key > Go To Next.
To loop through all the keys, select any of these menu selections repeatedly.

```

=====
Expressions
executed in each frame
in MEL command line
    expression -s "Object1.tx = sin(time)"
=====

```

SCRIPT EDITOR

use icon (window w/ horizontal dividing line) in lower right of Maya window

or

Window->General Editors->Script Editor

sections: history, input

type commands in input area (use CNTRL-Return to execute or ENTER)

Edit->Clear All

<interactively create a NURBS sphere>

see what gets typed in output section of Script Editor

Edit->Clear ALL

type into input area of Script Editor

```
$k = 0;
for ($i=0; $i<10; $i++) {
    sphere;
    $j = $i*2;
    move -r 0 $j 0;
}
$k = $k + 2;
```

SHELF BUTTON

highlight all but first commands & CNTRL-click drag to shelf to make it a button

hit button several times

(NOTE: MMB drag to trash can to remove)

SCRIPT FILE

make text file testScript.mel:

in Script Editor window

File->Source Script...

(NOTE: also facilities to create a UI; not worry about for now)

ATTRIBUTES

quantity of a node - e.g., sphere's transform node 'Rotate X'

nodename.attributename: e.g. mySphere.translateY or mySphere.ty

MEL commands: listAttr, getAttr, setAttr, aliasAttr

=====

```
sphere;
move -r 10 20 10;
getAttr nurbsSphere1.translateY;
getAttr nurbsSphere1.tx;
listAttr nurbsSphere1;
setAttr nurbsSphere1.tz 5;
```

```
aliasAttr up nurbsSphere.ty;
getAttr nurbsSpherel.up;
```

```
-----
keyable v. non-keyable attributes: attributes that can be keyframed
Channel Control Editor: controls keyability of attributes
  Window->General Editors->Channel Control...
Channel Box: info about object's transform node, shape node, input node
  Display->UI Elements->Channel Box/Layer Editor
  displays keyable and displayed non-keyable attributes
  can edit keyable attributes
Attribute Editor
  edit more attributes of object e.g., change name of objects
  Window->Attribute Editor
add custom attributes e.g., weight
  Modify->Add Attribute
```

```
=====
Expressions in scripts
```

```
// http://www.fundza.com/mel/quickref2/#expression1
// NOTE: `sphere` returns 2 strings. that's why $obj is an array
// and why $obj[0] is used in the expression
```

```
string $exp = "";
string $obj[];

for($i = 0; $i < 3; $i++) {
  $obj = `sphere`;
  move (rand(-3,3)) (rand(-3,3)) (rand(-3,3));
  $exp += "select -r " + $obj[0] + ";\n" +
    "move -moveY (rand(0,2));\n";
}
$exp += "select -clear;\n";

expression -s $exp -ae 1;
playbackOptions -min 1 -max 30;
play;
```

```
-----
Edit, manage expressions
Window->Animation Editors->Expression Editor...
select object, select expression, edit, then hit 'Edit'
in Help, see "Differences between expression and MEL syntax"
```