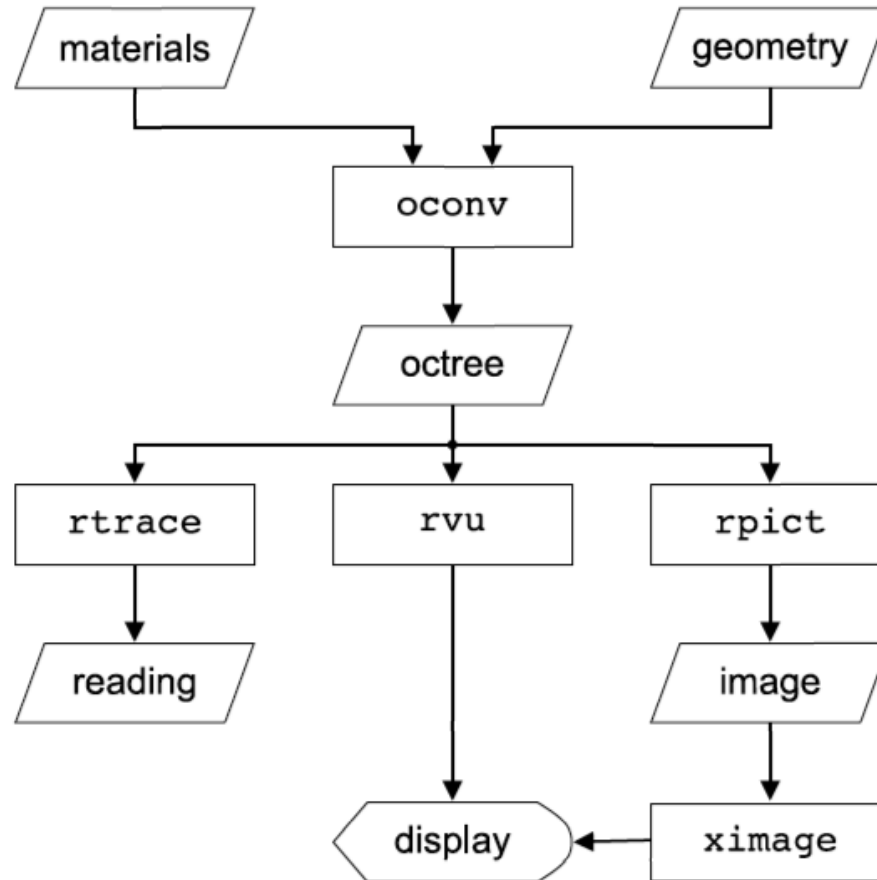# Radiance Scripting Toolkits
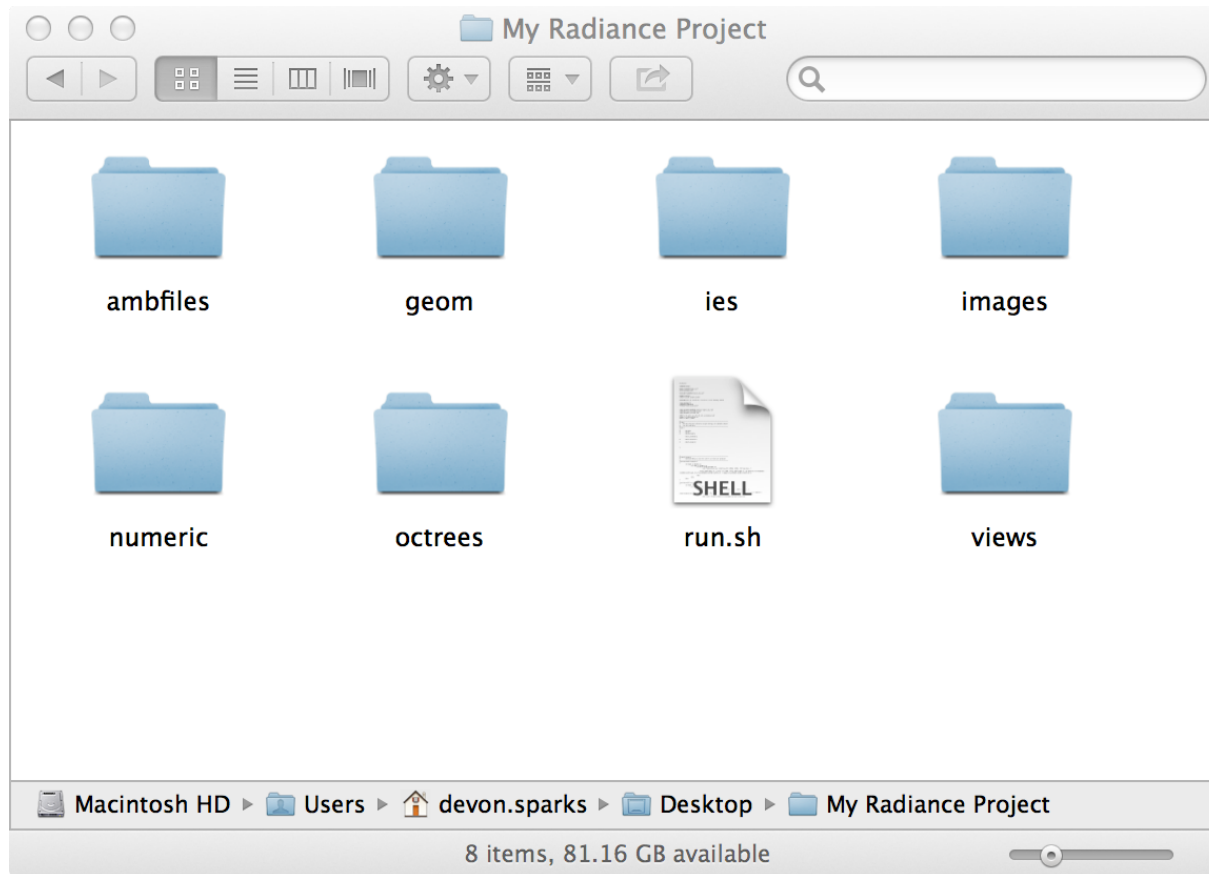
Workflows, Experiments, Opportunities

Devon Sparks | Arup Lighting

The Radiance Worldview

My Workflow | File Structure

My Workflow | Script Structure

```
#--------------------------------------------------------------
# main
#   The main function called on script startup; all subtasks should
#     be run from here.
#--------------------------------------------------------------
main()
{
    mkclean;
    mkrads;
    mkroom_lights;

    mkall_roommodels;

    mkall_roomcalcs;

    mkall_roompics;

}
```

ARUP

My Workflow | Utilities

```
#------------------------------------------------------------
# mkrads
#   Creates all the rad files we need give a directory of obj files
#------------------------------------------------------------
function mkrads()
{
    echo "(Re)generating .rad files...\n"
    for f in $GEOMDIR/*.obj; do
        obj2rad -f $f > $f.rad
    done
}
```

ARUP

My Workflow | Utilities

```
#---------------------------------------------------------------
# mkclean
#   Remove all the files created by this script
#---------------------------------------------------------------
function mkclean()
{
    echo "Removing old working files...\n"
    rm ambfiles/*.*
    rm model/*.rad
    rm octs/*.oct
    rm images/*.hdr
}
```

ARUP

My Workflow | Basic Parametrics

```
#-----------------------------------------------------------
# mkall_roompics
#    Generate HDR pics once the room-* octtrees are produced
#-----------------------------------------------------------
function mkall_roompics()
{
    for FLOOR in $FLOORS; do
        for WALL in $WALLS; do
            for SP in $SPACING; do
            echo "Generating room rendering with ($FLOOR, $WALL, '$SP spacing)..."

            vwrays $VIEW $RES -ff | rtrace -ffc $AMB `vwrays $RES $VIEW -d` -af
                ambfiles/room-R${FLOOR}-C${WALL}-S${SP}.amb octs/room-R${FLOOR}-C$
                {WALL}-S${SP}.oct > images/room-${FLOOR}-${WALL}-S${SP}.hdr &

    done; done; done
}
```

My Workflow | Some Shell Frustrations

Abstraction **Functions can't easily return values**

Composition **Data structures are weak**

Computation **Math is outsourced (*expr, bc, rcalc, etc.*)**

Control Flow **N-ary cartesian products aren't easy;**
                Higher-order functions aren't friendly.

ARUP

Experiment #1 | Stronger Shells

<ESDemo>
Local Variables
Higher order functions
Sane Parametrics

ARUP

Experiment #2 | Other Worldviews

[Radiance-general] Leveraging the Python language in Building Performance Simulation

Thomas Bleicher tbleicher at googlemail.com
Mon Dec 3 17:41:31 PST 2012

- Previous message: [Radiance-general] Leveraging the Python language in Building Performance Simulation
- Next message: [Radiance-general] Leveraging the Python language in Building Performance Simulation
- Messages sorted by: [date] [thread] [subject] [author]

```
Marcus

You can find some advanced scripts on Francesco's web site:

http://www.bozzograo.net/radiance/index.php?module=Downloads&func=view&cid=2&start=0

These are complete scripts that create new features on top of the Radiance
tool set. If you don't mind digging a bit you can look for the b/rad script
on his web site. It is a Radiance exporter for the 3D modeller Blender
(outdated though, won't work with the current Blender version). This
exporter has also a lot of import features and so the scripts modules show
how to read Radiance files.

As far as an "official" Radiance Python module goes there was some
discussion about it quite a while ago but that pretty much was it. I wrote
(or re-wrote) the base image class for wxfalsecolor specifically to allow a
reuse in other scripts but I don't think anyone has ever used it. I think
one of the reasons is that with Radiance even trivial scripting can go a
long way in terms of automation. Most scripts will be written quick and
dirty and don't need big complicated wrappers to serve their purpose.
Radiance encourages this because it's composed of numerous command line
tools, each with a simple task.

I don't know too much about EnergyPlus and the options to use it on the
command line. Perhaps there is not much to do about the actual calculation
part of the process and the automation comes in when the input is created
or the output is processed. In both cases a parser library for the file
formats used by EP would come in quite handy.

Regards,
Thomas
```
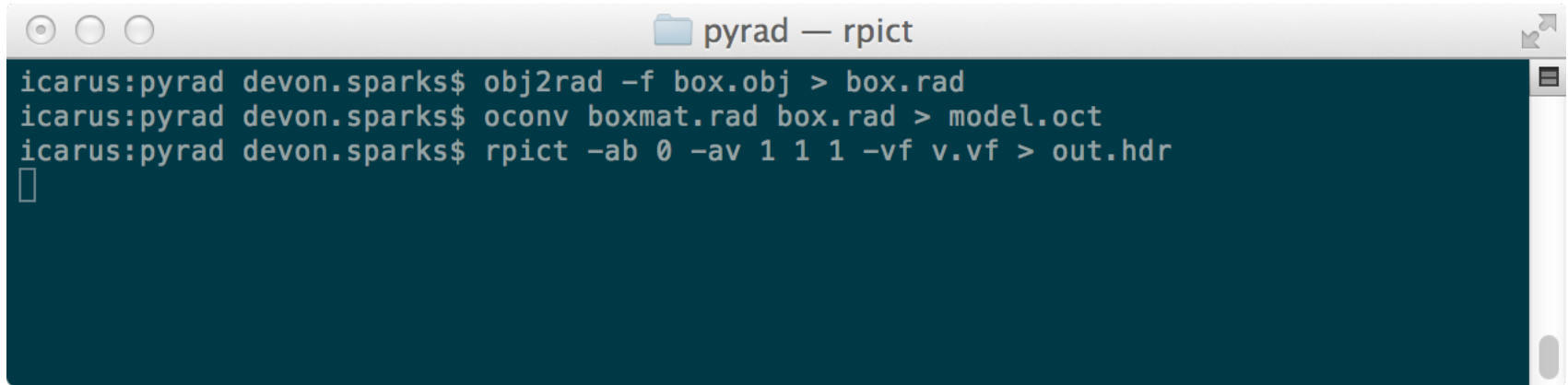
ARUP

Experiment #2 | Other Worldviews

<PyDemoProject>

ARUP

Experiment #2 | Other Worldviews

```
icarus:pyrad devon.sparks$ obj2rad -f box.obj > box.rad
icarus:pyrad devon.sparks$ oconv boxmat.rad box.rad > model.oct
icarus:pyrad devon.sparks$ rpict -ab 0 -av 1 1 1 -vf v.vf > out.hdr
```

pyrad — rpict

ARUP

Experiment #3 | Empowered Environments

<EnvDemos>
Script Templates as Services
Folder Actions

ARUP

Experiment #4 | Graphical Pipes

Opportunities | A Wishlist

– Strong abstraction and composition facilities (i.e. grouping related operations, storing compound data)

– Inherit the best of shells (directory traversal, integration with existing Radiance worldview)

– Language independence; text as a universal interface

– Sugar for bitter pills (Automating Revit model processing)

ARUP

Questions?
Ideas?

ARUP