Process of Building and Testing OpenStudio for Release

# System Requirements

Windows machine with configured OpenStudio build environment (<https://github.com/NREL/OpenStudio/wiki/Configuring-OpenStudio-Build-Environments>)

SmartGit

VMWare Workstation

Mac VM

Ubuntu VM

Incredibuild

S3

Sketchup

# Initial Steps

Connect to developer VPN (currently increases speed of uploads).

With SmartGit, merge develop into iteration.

# Builds

## Ubuntu

In SmartGit, pull iteration branch.

In a command window:

cd openstudio/build

ccmake ..

In CMake check the following:

BUILD\_PACKAGE

BUILD\_SWIG

CMAKE\_BUILD\_TYPE = Release

CMAKE\_BINARY\_DEB

In CMake type the following:

c

g

In a command window:

make –j8

make package\_deb

Copy build from VM to Windows

## Windows 64 bit

In SmartGit, switch to iteration branch and pull.

In CMake, select current 64 bit compiler

In CMake check the following:

BUILD\_CSHARP\_BINDINGS

BUILD\_DOCUMENTATION

BUILD\_PACKAGE

BUILD\_SWIG

BUILD\_TESTING

In CMake type the following:

c

g

In Visual Studio:

Open OpenStudio.sln

Select Release Solution Configuration

Build OpenStudio until “configuring done”, and “generating done”

Open OpenStudioCore.sln

Select Release Solution Configuration

Build OpenStudioCore with Incredibuild

## Windows 32 bit

In SmartGit, switch to iteration branch and pull.

In CMake, select current 32 bit compiler

In CMake check the following:

BUILD\_CSHARP\_BINDINGS

BUILD\_PACKAGE

In CMake type the following:

c

g

In Visual Studio:

Open OpenStudio.sln

Select Release Solution Configuration

Build OpenStudio until “configuring done”, and “generating done”

Open OpenStudioCore.sln

Select Release Solution Configuration

Build OpenStudioCore with Incredibuild

## Mac

In SmartGit, pull iteration branch.

In a command window:

cd openstudio/build

ccmake ..

In CMake check the following:

BUILD\_PACKAGE

In CMake type the following:

T

In CMake **uncheck** the following:

CPACK\_BINARY\_FOO (and uncheck all)

In CMake **check** the following:

CPACK\_BINARY\_PACKAGEMAKER

In CMake type the following:

c

g

In a command window:

make package –j8

<ctrl> c

Windows’ button q (to quit a Mac app)

Copy build to VM’s share folder

# GitHub Releases

<https://github.com/NREL/OpenStudio/releases>

Select “Draft a new release”

Set tag version = vX.Y.Z

Set Target = iteration

Set Release title = OpenStudio vX.Y.Z

Check “This is a pre-release” for an iteration build

Select “Save draft”

Publish release

Attach binaries by drag and drop (tip: Save draft after each binary successfully added)

Select “Publish release”

# OpenStudio.net

In S3:

In “openstudio-builds”, make bucket for new builds (X.Y.Z)

Drag and drop the builds into above bucket

Right-click on the newly uploaded files, and create URLs

In OpenStudio.net:

(<https://www.openstudio.net/user>)

Select “Edit OpenStudio Release Links”

Update “Current Release Version” or “Develop Release Version” (depending on whether a major or iteration build is being done)

Replace the S3 build URLs with those created above

# Documentation

In CMake check the following:

BUILD\_DOCUMENTATION

In Visual Studio:

Open OpenStudio.sln

Build OpenStudio in release until “configuring done”, open OpenStudioCore in release

Open OpenStudioCore.sln

**Without** Incredibuild, build ALL\_DOCYGEN

**Without** Incredibuild, build ALL\_RDOC

In Windows Explorer:

In folder C:\Git\OpenStudio\_1\build\OSCore-prefix\src\OSCore-build\doc

Extract zip OpenStudio-X.Y.Z-doc.zip to a similarly named folder

In S3:

Drag folder into S3 bucket “openstudio-sdk-documentation/cpp”

For major releases, delete the content of openstudio-sdk-documentation/cpp/latest,

and drag in the content of C:\Git\OpenStudio\_1\build\OSCore-prefix\src\OSCore-build\doc\OpenStudio-1.9.0-doc

In Windows Explorer:

In folder C:\Git\OpenStudio\_1\build\OSCore-prefix\src\OSCore-build\ruby\rdocify

Extract zip OpenStudio-1.x.x-rdoc.zip to a similarly named folder

In S3:

Drag folder into S3 bucket “openstudio-sdk-documentation/ruby/”

From S3 documentation bucket, download index.htm and edit to new release number (23 changes)

Test at <https://openstudio-sdk-documentation.s3.amazonaws.com/index.html>

# OSVersion Testing

In Windows Explorer:

In folder C:\Git\OpenStudio\_1\build\OSCore-prefix\src\OSCore-build\Products\Release open cmd prompt

Drag and drop openstudio\_osversion\_tests.exe onto cmd prompt, run, and make sure everything passes

# Sanity Testing Release Builds

TBD

# Version Update

In Windows Explorer:

In C:\Git\OpenStudio\_1\build\OSCore-prefix\src\OSCore-build\resources\osversion copy folder X.Y.Z to C:\Git\OpenStudio\_1\openstudiocore\resources\osversion (new folder, 3 files)

In C:\Git\OpenStudio\_1\ update CMakeList version (I line)

In C:\Git\OpenStudio\_1\openstudiocore\resources update CMakeList version (3 lines, 1 location)

In C:\Git\OpenStudio\_1\openstudiocore\resources\model update OpenStudio.idd version (1 line)

In C:\Git\OpenStudio\_1\openstudiocore\src\osversion update VersionTranslator.cpp version

At <https://github.com/NREL/OpenStudio>

Type t

Type VersionTranslator.cpp

Select “history”, see edits if needed (usually use defaultUpdate, 2 lines)

In SmartGit:

Commit above files (Commit Message = “Updating version to X.Y.Z”)

# AMI BUILD

In SmartGit:

Pull OpenStudio-server master

In Windows Explorer:

Update version in C:\Git\OpenStudio-server\CHANGELOG.md

Update version in C:\Git\OpenStudio-server\vagrant\chef\roles\openstudio.rb

Update version in C:\Git\OpenStudio-server\server\lib\openstudio\_server\version.rb

In SmartGit:

Commit above files

In a command window:

In C:\Git\OpenStudio-server, run “rake release”

# Compatibility Matrix

<https://github.com/NREL/OpenStudio/wiki/OpenStudio-Version-Compatibility-Matrix>

Select “Edit” and add new row of information