Rapid Application Development with OpenStudio

Evan Weaver, Nicholas Long, Katherine Fleming, Marjorie Schott, Kyle Benne, Elaine Hale
National Renewable Energy Laboratory[1]

ABSTRACT

This paper presents several case studies of rapidly implemented, audience-specific applications for whole building energy modeling and standards analysis. By tailoring each application to the audience and the task at hand, the required learning curve for new users was greatly reduced. Each case study used OpenStudio, the U.S. Department of Energy's middleware software development kit. OpenStudio provides an easy interface to the EnergyPlus whole building simulation engine, extends its capabilities, and provides higher level functionality such as software interoperability, standards, analysis, and optimization.

Each case study is unique in the technology employed to interface with OpenStudio as well as in its methods for user interaction and data presentation. Four case studies are presented: (1) a visually rich web-based application that provides a straightforward method to determine if proposed building modifications will qualify for federal tax deductions; (2) a web application that uses user scripts and requests high-level building and weather data and enables the user to quickly construct new EnergyPlus input files; (3) a desktop application that provides a well-structured workflow for novice building modeling practitioners while they perturb building models and evaluate the effects of those changes on codes and standards compliance; and (4) an intuitive heating, ventilation, and air-conditioning system modeling tool that was extended to become a whole building energy modeling application. Finally, this paper discusses the design process employed during the creation of each application as an aid to future developers of OpenStudio-based applications.

Introduction

Whole building energy modeling is becoming increasingly prevalent. Tools such as EnergyPlus (Crawley et al. 2008) provide whole building energy prediction capability, and the computational power available in current computers makes analysis of large numbers of models less burdensome. Energy prices compel building owners to reduce energy consumption; government regulations continually mandate increasing building energy efficiency. Whole building energy modeling, performed early in the design phase, allows building designers to optimize the sizing of building mechanical systems, envelope and fenestration composition, and siting and building form; they can also accurately predict the most cost-effective energy conservation measures (ECMs).

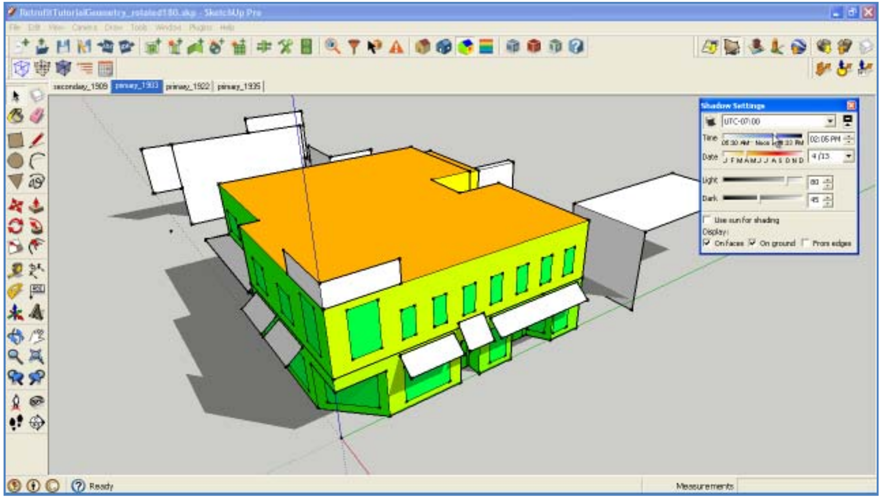
[1] The Alliance for Sustainable Energy, LLC (Alliance), is the manager and operator of the National Renewable Energy Laboratory (NREL). Employees of the Alliance for Sustainable Energy, LLC, under Contract No. DE-AC36-08GO28308 with the U.S. Dept. of Energy, have authored this work. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for United States Government purposes.

An example of the value of whole building energy modeling was demonstrated during the design phase of the National Renewable Energy Laboratory's (NREL) Research Support Facility (RSF). This project had very aggressive energy use goals, and the design-build team was required to demonstrate design compliance both pre- and post-occupancy. Thus, simulation was used early and often throughout the design process (Guglielmetti et al. 2010). A comparison of predicted (modeled) energy consumption to actual energy consumption during the first year of building occupancy indicated the initial simulation predictions were extremely accurate. (Pless et al. 2011) Regular (and early) performance modeling enabled designers to build a structure that is now Leadership in Energy and Environmental Design (LEED) Platinum certified, provides excellent occupant comfort, is one of the most energy-efficient office buildings in the country, and was constructed at a cost comparable to typical code-compliant new commercial construction in the area. The role that modeling played in helping this project meet all the energy use goals cannot be overstated (Hirsch et al. 2011).

EnergyPlus (http://apps1.eere.energy.gov/buildings/energyplus) is a robust whole building energy analysis tool, and is available for free. It is feature rich and can model the integration of many unique building systems; however, it does not provide a graphical user interface (GUI). Thus, it is difficult for the novice modeler to use. Radiance (Ward 1994) is a physically based light-backwards raytracer that can simulate lighting and daylighting much more accurately than the light transport algorithms of EnergyPlus. This higher fidelity was required on the RSF project to accurately capture the interactions of the lighting and the building mechanical systems. Unfortunately, Radiance, like EnergyPlus, lacks a GUI interface, making it somewhat impenetrable for the novice user. In addition, merging the simulation output data from the two programs is a complex problem (Guglielmetti et al. 2010).

OpenStudio (http://openstudio.nrel.gov) was initially created as a simple GUI for EnergyPlus (Ellis et al. 2008) leveraging Google SketchUp to bring building geometry into EnergyPlus (Figure 1). In December 2010, OpenStudio was relaunched and expanded with the goal of establishing an object-oriented framework for building energy modeling, providing a way to holistically leverage existing building simulation tools (Guglielmetti 2011). Supporting methods in the software development kit (SDK) provide access to model object attributes and enable programmatic articulation and modification (perturbation) of models. The application programming interface (API) for OpenStudio extends the EnergyPlus input data file (IDF) functionality, enables interoperation with the Radiance lighting simulation engine, and provides the capability for rapid software development of command line scripts (Ruby and C# are natively supported), or fully formed GUI applications.

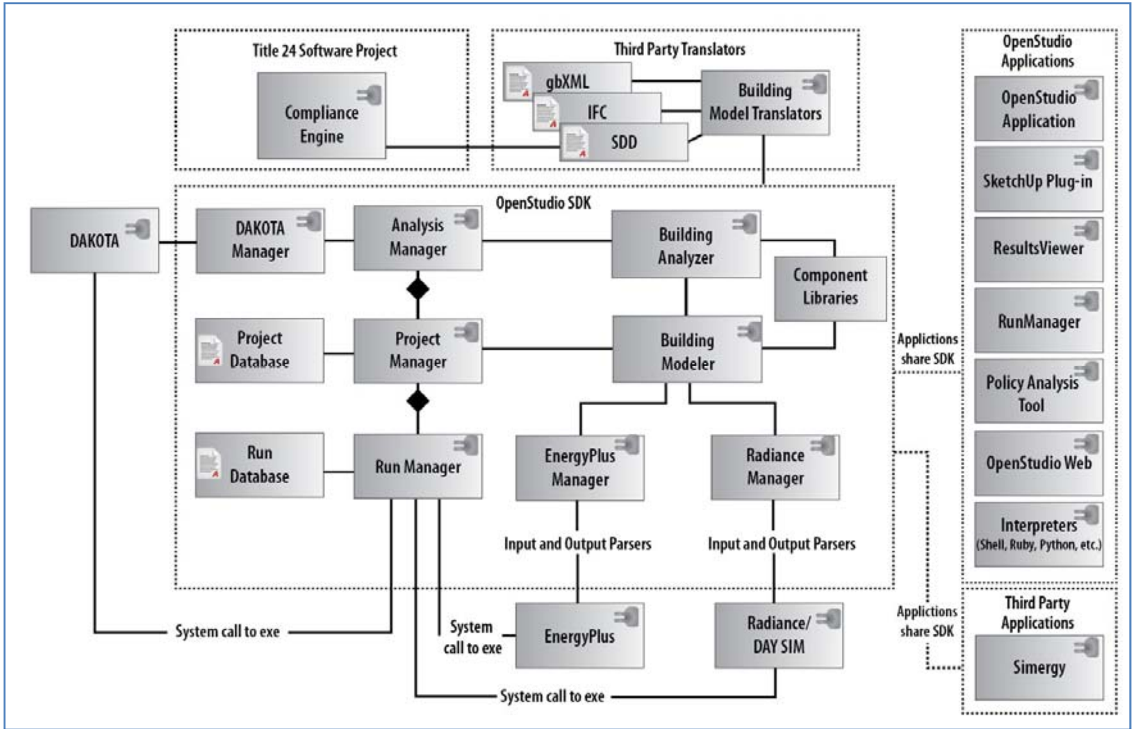
Figure 1. The OpenStudio Plug-in™ for Google SketchUp



Credit: David Goldwasser/NREL

OpenStudio is an open-source, cross-platform, cross-language SDK and application suite. Windows, Mac, and Linux users can obtain all source code for free and, following the lesser general public license (LGPL) protocol, use C++, C#, or Ruby programming languages to extend functionality or create entire new applications that leverage NREL's OpenStudio work. The OpenStudio architecture is shown in Figure 2, and illustrates simulation engine interoperability, translators for third-party formats, and example applications that are shipped as part of OpenStudio.

Figure 2. OpenStudio SDK Supports Multiple Simulation Engines and Input Formats to Facilitate Application Development

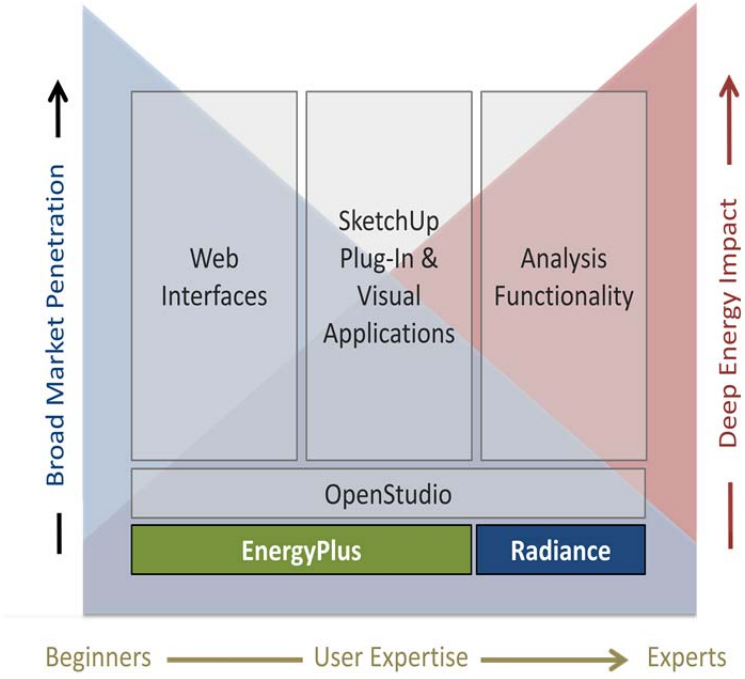


Credit: Elaine Hale/NREL

OpenStudio provides an SDK and a comprehensive suite of example building modeling applications that leverage the SDK. C# and Ruby bindings to the native OpenStudio C++ source enable rapid prototyping of new functionality without compiling from source, and frameworks are provided to integrate these languages into example applications. By itself, the OpenStudio application suite—a series of demonstration applications built with the OpenStudio SDK—makes building energy modeling easier and more accessible to potential practitioners. The SDK provides flexibility for developers to either build on these applications or create entirely new ones to address the diverse needs of the market. The SDK provides easy interfaces to EnergyPlus and Radiance, and allows developers to quickly develop a tailored solution for a specific target audience, from novice whole building energy modeler, to expert (Figure 3).

To provide concrete, reusable, and extensible examples of OpenStudio's SDK use, we provide four case studies that discuss web and desktop applications that leverage the SDK. The order of presentation is essentially chronological, and indicates the continually increasing capability and richness of the OpenStudio SDK.

Figure 3. OpenStudio Makes EnergyPlus and Radiance More Readily Available to a Diverse Audience



Credit: Larry Brackney/NREL

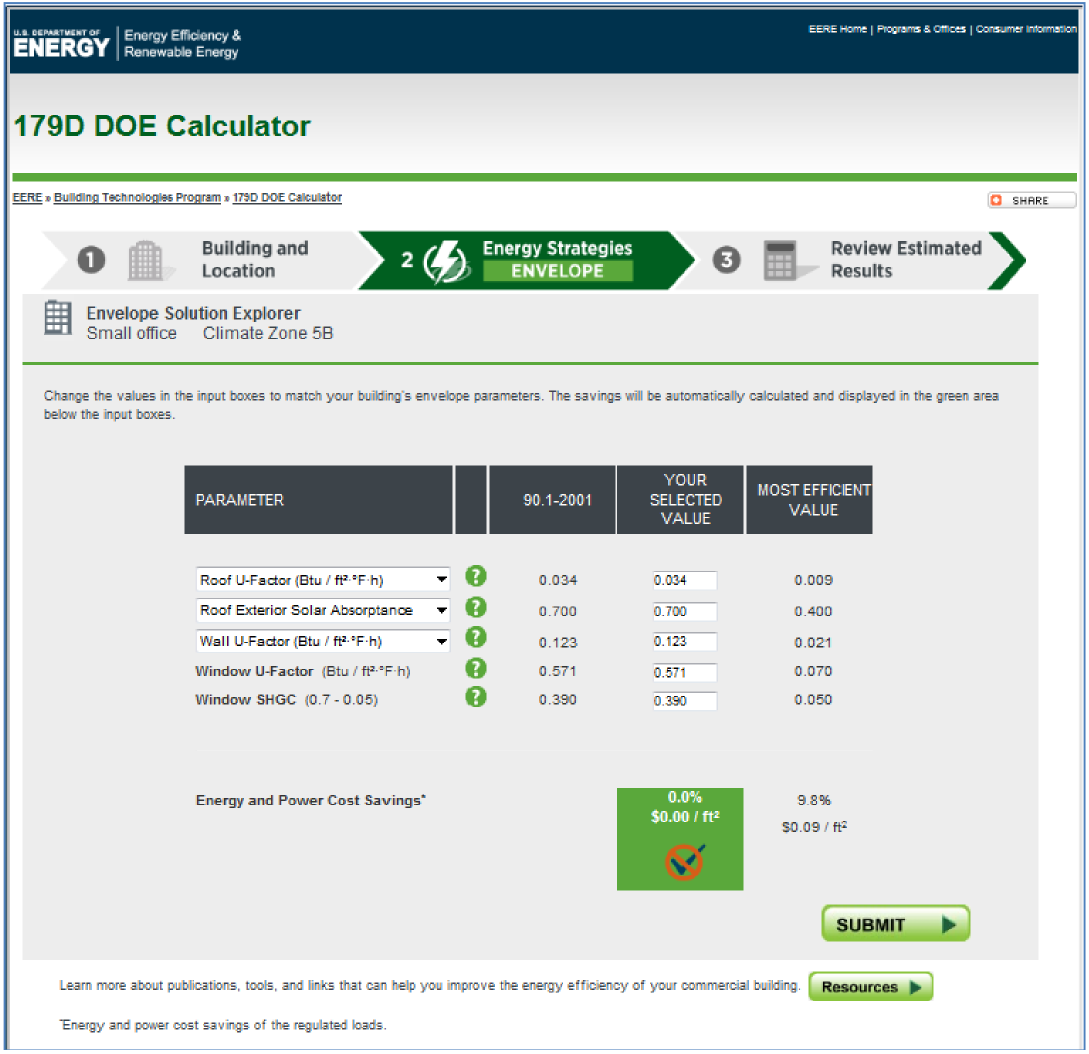
Case Study I: 179D DOE Calculator

The federal government has established tax incentives to encourage commercial building owners to improve the energy efficiency of their buildings, and the U.S. Department of Energy (DOE) was tasked with raising public awareness of these incentives. Detailed energy analysis—often required to determine qualification for the energy efficiency tax incentives—is often deemed too expensive for single building owners. A high-level, easily accessible tool was needed to help owners decide whether they might qualify for the 179D tax deduction. With that in mind, the web-based 179D DOE Calculator (Figure 4) was created. (http://apps1.eere.energy.gov/buildings/commercial_initiative/179d/)

The 179D DOE Calculator uses several technologies:

- ColdFusion – a web programming language and database access layer
- Ruby – an open-source scripting language used to expose OpenStudio functionality (http://ruby-lang.org)
- OpenStudio – The SDK subject of this paper
  - Building Model – OpenStudio's data abstraction representing a fully articulated model
  - Project – The OpenStudio project level database for data serialization
  - RunManager – OpenStudio's SDK and toolset for queuing and processing discrete work units that may include EnergyPlus or Radiance simulations along with scripted analysis
- EnergyPlus – DOE's whole building, open-source energy simulation software

Figure 4. 179D DOE Calculator


U.S. DEPARTMENT OF ENERGY | Energy Efficiency & Renewable Energy
EERE Home | Programs & Offices | Consumer Information
SHARE
1 Building and Location
2 Energy Strategies ENVELOPE
3 Review Estimated Results
SUBMIT
Resources

179D DOE Calculator

EERE » Building Technologies Program » 179D DOE Calculator

Envelope Solution Explorer
Small office   Climate Zone 5B

Change the values in the input boxes to match your building's envelope parameters. The savings will be automatically calculated and displayed in the green area below the input boxes.

PARAMETER		90.1-2001	YOUR SELECTED VALUE	MOST EFFICIENT VALUE
Roof U-Factor (Btu / ft^2·°F·h)	?	0.034	0.034	0.009
Roof Exterior Solar Absorptance	?	0.700	0.700	0.400
Wall U-Factor (Btu / ft^2·°F·h)	?	0.123	0.123	0.021
Window U-Factor (Btu / ft^2·°F·h)	?	0.571	0.571	0.070
Window SHGC (0.7 - 0.05)	?	0.390	0.390	0.050
Energy and Power Cost Savings*			0.0% $0.00 / ft^2	9.8% $0.09 / ft^2

Learn more about publications, tools, and links that can help you improve the energy efficiency of your commercial building.

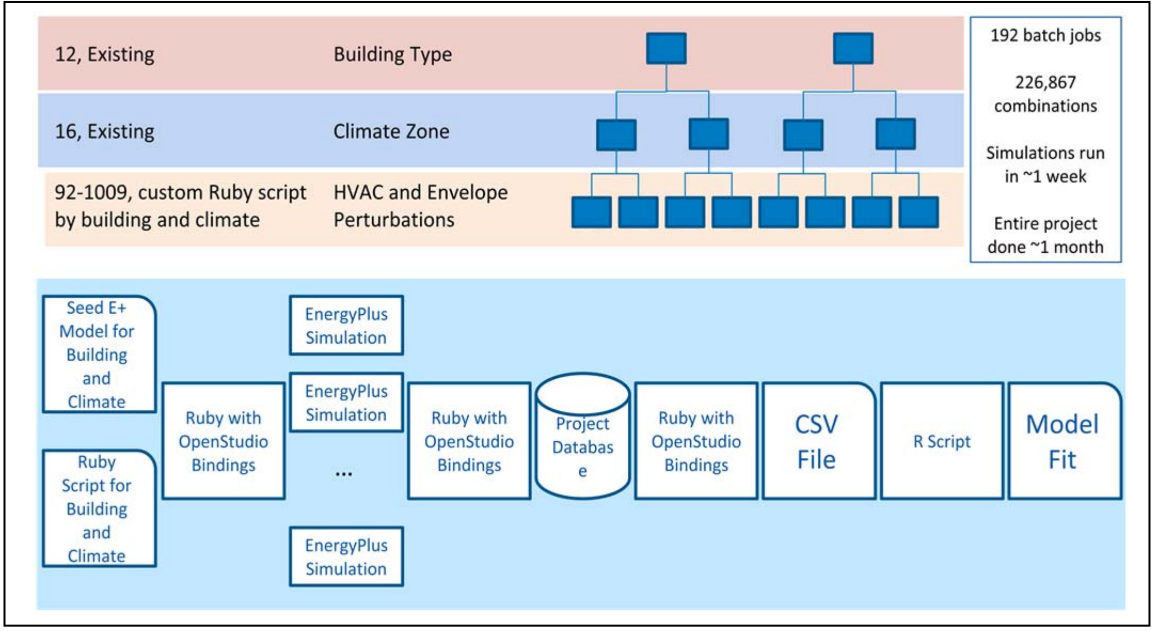
*Energy and power cost savings of the regulated loads.

Credit: Evan Weaver/NREL

The underlying premise of the 179D DOE Calculator (Deru et al. 2012) is that a very large database of pre-computed simulations associated with multiple building archetypes, climate zones, and a range of design alternatives may be accessed via simple web forms. These simulation results are compared with performance targets, the user is apprised of the potential to earn the 179D tax deduction for a particular building configuration, and additional instructions are provided for the application. OpenStudio scripting and run management were key to creating the supporting database of simulation results in a timely manner. The analysis process is shown in Figure 5.

The 179D DOE Calculator evaluates the effects of ECMs applied to building envelope, lighting, or heating, ventilation, and air conditioning (HVAC). To cover a broad range of scenarios, 12 DOE reference building models (Field et al. 2010) and 16 climate zones were used as baseline models. Ruby scripting exposed OpenStudio functionality to create and perturb energy models, which were simulated using OpenStudio's RunManager. NREL's and Sandia National Laboratories' high-performance computing (HPC) resources—including the Red Mesa and Red Rock super computers—executed approximately 250,000 simulations. Simulations and associated results were stored in OpenStudio's project database for use in generating regression equations via the R scripting language (http://www.r-project.org). Only the regression equation coefficients were required by the website, providing a highly responsive user interface with a logical workflow to easily guide new users. This project involved 6 staff members, yet required just 8 calendar weeks from initiation to web launch.

Figure 5. OpenStudio Enabled 179D Analysis Process


12, Existing
Building Type
16, Existing
Climate Zone
92-1009, custom Ruby script by building and climate
HVAC and Envelope Perturbations
192 batch jobs
226,867 combinations
Simulations run in ~1 week
Entire project done ~1 month
Seed E+ Model for Building and Climate
Ruby Script for Building and Climate
Ruby with OpenStudio Bindings
EnergyPlus Simulation
EnergyPlus Simulation
...
EnergyPlus Simulation
Ruby with OpenStudio Bindings
Project Databas e
Ruby with OpenStudio Bindings
CSV File
R Script
Model Fit

Credit: Dan Macumber/NREL

Case Study II: Ruby on Rails Scripting

OpenStudio (typified by the Plug-in for Google SketchUp) has been often thought of as a client application running on a single computer with one user; however, OpenStudio has been directly integrated into web applications as well. "OpenStudio Live" is a simple web-based application that leverages the OpenStudio SDK via a web interface and accepts high-level building parameters to fully articulate a building energy model entirely via web services.

The web application was built using Ruby on Rails (http://rubyonrails.org). The OpenStudio SDK was compiled on a Red Hat Enterprise Linux web server and packaged into a RubyGem that is installed into a Rails Bundle (http://rubyforge.org/projects/rubygems, http://gembundler.com/rails3.html). Using Rails' rapid prototyping capabilities allowed the application to be generated in 2 days as a proof of concept. The connection to OpenStudio from Rails is performed entirely through the RubyGem, which exposes all the SDK's methods to Rails.

Ruby on Rails is designed to communicate easily with other APIs. OpenStudio Live exercises an API to the publically available Building Component Library (BCL) (Long et al. 2011)—the DOE's comprehensive online database of building components, which can be accessed and used directly by the OpenStudio SDK—to select the closest weather file based on the high-level input data. Search data returned by the BCL API to OpenStudio Live produces a list of closest results, and allows the user to select a specific weather file. Once selected, another API call to the BCL downloads, parses, and inserts the appropriate content into the OpenStudio model.

Security is of great concern in web applications. OpenStudio running as a Gem in Rails poses minimal threat because OpenStudio runs entirely in memory and the data being sent to OpenStudio are not persisted (saved). However, buffer overruns still present a concern, and software unit tests are written to ensure that data buffering and pointers are managed appropriately by the OpenStudio core code. This same level of diligence is required when leveraging OpenStudio's extensibility via user scripts.

Although a simple example of a web application which uses OpenStudio, it demonstrates the power of the OpenStudio SDK and presents a logical approach for replacing the EnergyPlus Example File Generator (http://apps1.eere.energy.gov/buildings/energyplus/cfm/inputs/).

This project involved 2 staff members, and required 2 days.

Case Study III: Policy Analysis Tool

To make commercial building codes and standards compliance testing accessible to a broad audience, a high-level, intuitive GUI presenting basic modeling functionality for beginning users was needed. Functional requirements included selection of building types, climate zones, perturbations (including lighting and envelope), and a limited set of code compliance rules for assessing perturbations. The end result application managed simulations across multicore computers, stored simulation output in a local database, and enabled the user to visualize and compare the energy and cost performance of results. Application robustness, user interface intuitiveness, and data presentation were of primary importance.

The Policy Analysis Tool is a data manipulation and data visualization tool related to building model perturbations and the application of codes and standards for commercial buildings. It has a GUI that provides a clear and intuitive workflow, simplifying the process of data generation and display of results. The example application enabled demonstration of several key pieces of OpenStudio functionality including management of perturbations, rulesets, multiple simulations within a "project," establishment of a workflow-centric user interface template that has been leveraged in subsequent OpenStudio applications, and integration of methods for output visualization.

As with all OpenStudio example applications, the Policy Analysis Tool is open-source and cross-platform. It leverages EnergyPlus, the OpenStudio SDK, and the following libraries:

- Boost (http://www.boost.org) – C++ development libraries
- Qt (http://qt.nokia.com) – GUI development framework
- VTK Charts (http://www.vtk.org/Wiki/VTK/Charts) – scalable OpenGL data charting
- C Language Integrated Production System (CLIPS) (http://clipsrules.sourceforge.net) – building expert systems rules processing

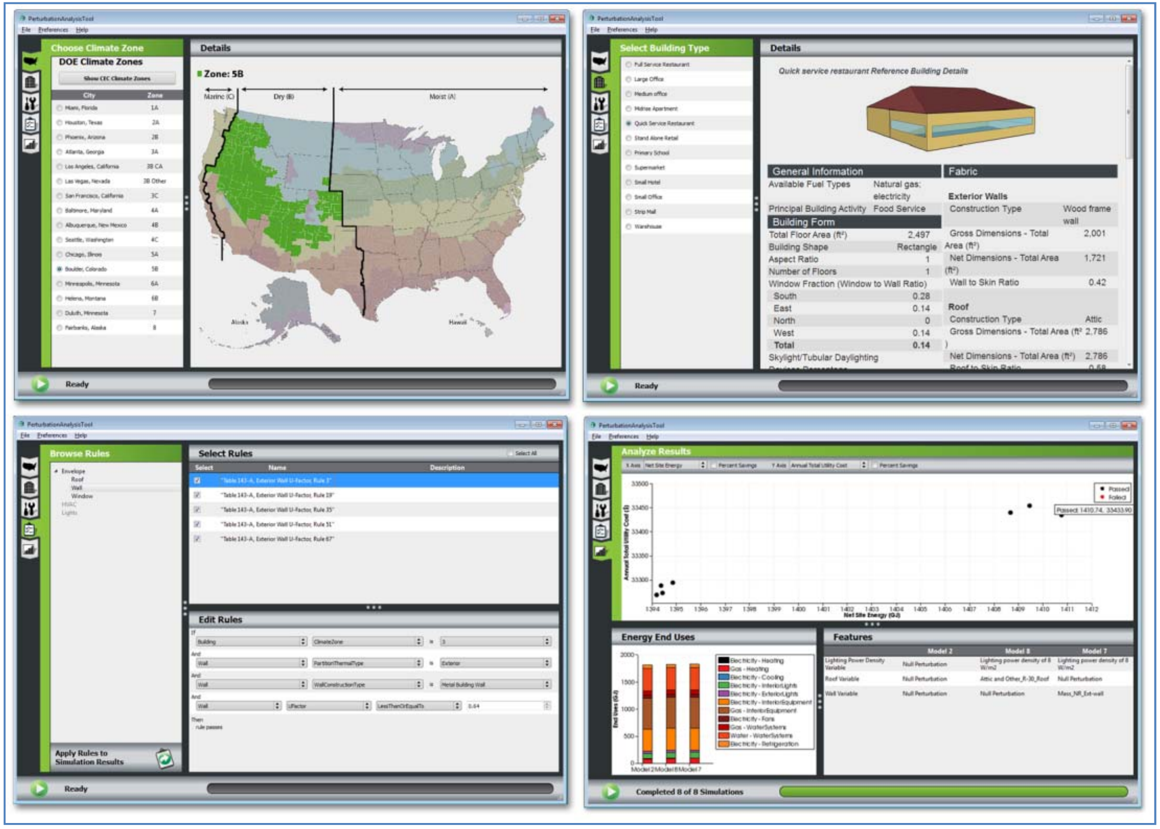
It is particularly worth noting that OpenStudio's graphical applications use Qt, an open-source, cross-platform, and feature-rich library for GUI development. Rapid application development is possible using intrinsic Qt features and NREL's expanding collection of OpenStudio-specific Qt widgets (modular GUI components) that expose the SDK's functionality and enable even faster application development. By using Qt as described, the required investment to build a fully featured GUI for whole building energy modeling is only a small fraction of the investment previously made in the OpenStudio SDK and EnergyPlus.

In addition to the aforementioned Project and RunManager components of the SDK, the Policy Analysis Tool leveraged these OpenStudio components:

- Perturbations – A portion of Project that enables manipulation of building models
- Ruleset – The section of the API responsible for expressing rules and ruleset options written using grammatical clauses composed of data dictionary terms, operators, and low-level (boolean, double, integer, and string) values
- RulesEngine – The runtime engine for applying rulesets to building energy models using the CLIPS expert shell

A tabbed workflow makes use more intuitive but it is not rigidly enforced, allowing the user to navigate at will. The workflow uses a tab sequence paradigm (Figure 6).

Figure 6. A Typical Policy Analysis Tool Workflow



DOE Climate Zones

City	Zone
○ Miami, Florida	1A
○ Houston, Texas	2A
○ Phoenix, Arizona	2B
○ Atlanta, Georgia	3A
○ Los Angeles, California	3B CA
○ Las Vegas, Nevada	3B Other
○ San Francisco, California	3C
○ Baltimore, Maryland	4A
○ Albuquerque, New Mexico	4B
○ Seattle, Washington	4C
○ Chicago, Illinois	5A
● Boulder, Colorado	5B
○ Minneapolis, Minnesota	6A
○ Helena, Montana	6B
○ Duluth, Minnesota	7
○ Fairbanks, Alaska	8

Zone: 5B

Quick service restaurant Reference Building Details

General Information	
Available Fuel Types	Natural gas; electricity
Principal Building Activity	Food Service
Building Form	
Total Floor Area (ft²)	2,497
Building Shape	Rectangle
Aspect Ratio	1
Number of Floors	1
Window Fraction (Window to Wall Ratio)	
South	0.28
East	0.14
North	0
West	0.14
Total	**0.14**
Skylight/Tubular Daylighting	

Fabric	
Exterior Walls	
Construction Type	Wood frame wall
Gross Dimensions - Total Area (ft²)	2,001
Net Dimensions - Total Area (ft²)	1,721
Wall to Skin Ratio	0.42
Roof	
Construction Type	Attic
Gross Dimensions - Total Area (ft²)	2,786
Net Dimensions - Total Area (ft²)	2,786

Select Rules

Select	Name	Description
☑	"Table 143-A, Exterior Wall U-Factor, Rule 3"	
☑	"Table 143-A, Exterior Wall U-Factor, Rule 19"	
☑	"Table 143-A, Exterior Wall U-Factor, Rule 35"	
☑	"Table 143-A, Exterior Wall U-Factor, Rule 51"	
☑	"Table 143-A, Exterior Wall U-Factor, Rule 67"	

Edit Rules

If
Building | ClimateZone | = | 3
And
Wall | PartitionThermalType | = | Exterior
And
Wall | WallConstructionType | = | Metal Building Wall
And
Wall | UFactor | LessThanOrEqualTo | 0.64
Then
rule passes

Features

	Model 2	Model 8	Model 7
Lighting Power Density Variable	Null Perturbation	Lighting power density of 8 W/m2	Lighting power density of 8 W/m2
Roof Variable	Null Perturbation	Attic and Other_R-30_Roof	Null Perturbation
Wall Variable	Null Perturbation	Null Perturbation	Mass_NR_Ext-wall

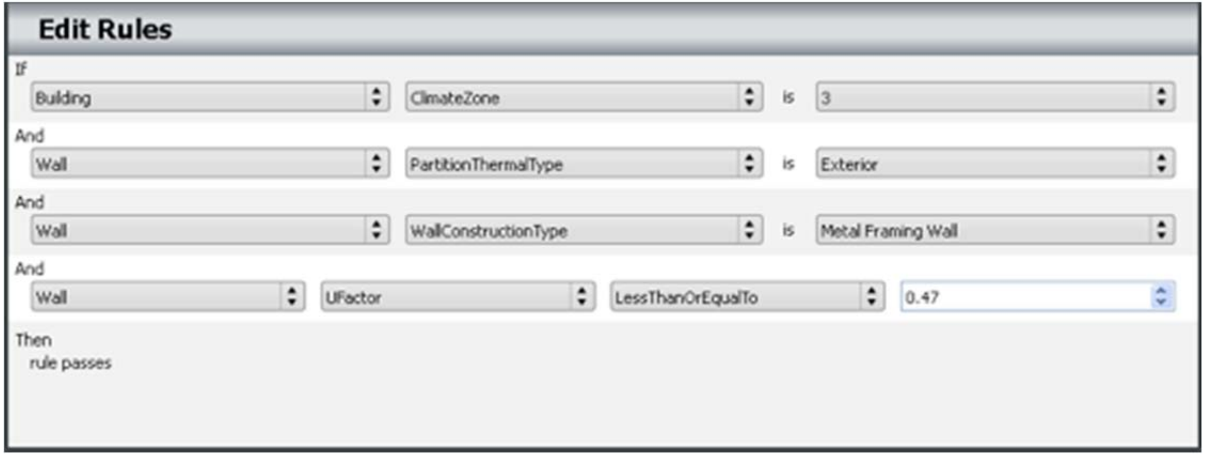
Credit: Marjorie Schott/NREL

Specific tab functionality includes:

- Climate Zone tab – The user selects, via a checkable button in the upper left pane, either a DOE/American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) climate zone map or a California Energy Commission (CEC) climate zone map. If the CEC climate zone map is chosen, a zoomed map is also available. Each climate zone map has 16 zones, any one of which is selectable via its respective radio button from a group in the lower left pane. As the radio buttons are selected, the corresponding geographical region is highlighted on the displayed map in the right pane.
- Building Type tab – The user selects, via a radio button from a group in the left pane, one reference building from a list of 12. Salient details about the selection are displayed in the right pane.
- Perturbations tab – The "Browse Perturbations" pane uses a tree view to present a hierarchical representation of the perturbations available in the database. Perturbations are grouped in a logical, predefined structure to aid in data presentation. The "Select Perturbations" pane uses a table view to show the perturbations associated with a selected node of the tree view to the left. The lower right pane uses a table view to display details associated with a perturbation selected in the table view. The user assembles a set of combinations by selecting checkboxes associated with perturbations in the table view.
- Rules tab – Currently, only Title 24 rules are supported, so the Rules tab is viewable only if a CEC climate zone is selected. The "Browse Rules" pane uses a tree view to present a

hierarchical representation of the rules in the database. Rules are grouped in a logical, predefined structure to aid in data presentation. The “Select Rules” pane uses a table view to show the rules associated with a selected node of the tree view to the left. The “Edit Rules” pane uses comboboxes and spinboxes to show details associated with a rule selected in the table view. The comboboxes and spinboxes provide syntactical structure for a rule, and allow the user to change parts of speech (Figure 7). Changes to the rules are saved to the project database once a simulation run is initiated, enabling the user to apply rulesets to simulation results by clicking a button at the bottom of the tree.

Figure 7. Policy Analysis Tool Interface for Visual Editing of Title 24 Compliance Rules



Edit Rules

If
Building ClimateZone is 3

And
Wall PartitionThermalType is Exterior

And
Wall WallConstructionType is Metal Framing Wall

And
Wall UFactor LessThanOrEqualTo 0.47

Then
rule passes

Credit: Elaine Hale/NREL

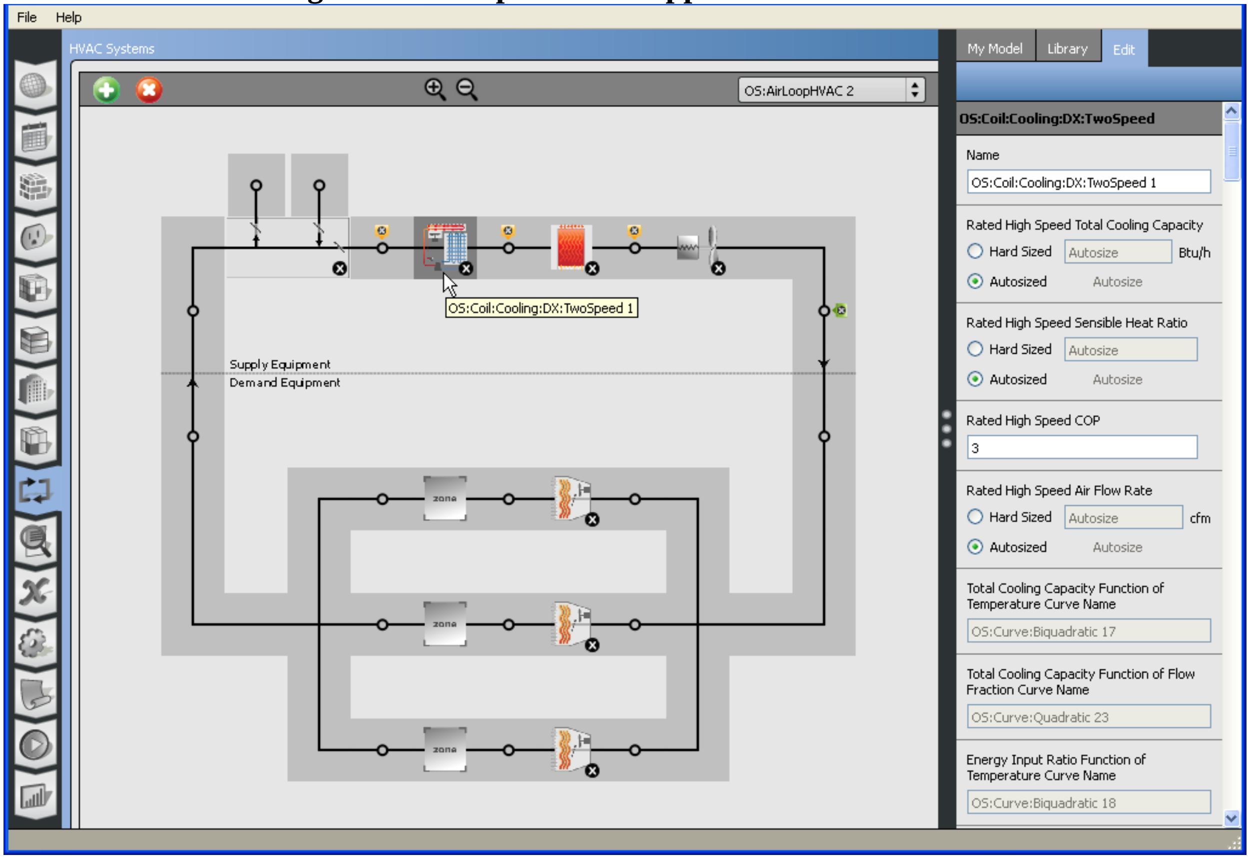
- Results tab – The results tab comprises 3 panes: “Analyze Results”, “Energy End Uses”, and “Building Features”. The “Analyze Results” pane shows the entire simulation set, quickly identifying buildings of interest based on criteria such as cost neutrality. Each point represents a different perturbation of the baseline building; a point’s color denotes whether it complies with the applied ruleset. Right-clicking a point adds it to the end-use chart view in the lower left or a perturbation summary table in the lower right. Panning and zooming functionality is provided with the left mouse button and scroll wheel, and the user may specify a variety of X- and Y-axis values.

The Policy Analysis Tool represents the first step toward developing a replacement tool for NREL’s Opt-E-Plus (Hale and Long 2010) that can be freely distributed for parametric analysis and optimization. This project involved 3 staff members and required 12 calendar weeks to produce from definition of requirements to software release.

Case Study IV: OpenStudio Application

The OpenStudio application suite has grown significantly, and now includes a number of example applications. As OpenStudio’s SDK functionality has grown and been exposed by GUIs, more building energy modelers have adopted the suite. Providing a GUI with a useful workflow that combines individual applications and encapsulates new SDK functionality was key to satisfying the needs of this growing audience of design and analysis practitioners, and simultaneously provides concrete demonstrations of expanding API capability.

Figure 8. The OpenStudio Application HVAC Tab



OS:Coil:Cooling:DX:TwoSpeed

Name
OS:Coil:Cooling:DX:TwoSpeed 1

Rated High Speed Total Cooling Capacity
○ Hard Sized Autosize Btu/h
● Autosized Autosize

Rated High Speed Sensible Heat Ratio
○ Hard Sized Autosize
● Autosized Autosize

Rated High Speed COP
3

Rated High Speed Air Flow Rate
○ Hard Sized Autosize cfm
● Autosized Autosize

Total Cooling Capacity Function of Temperature Curve Name
OS:Curve:Biquadratic 17

Total Cooling Capacity Function of Flow Fraction Curve Name
OS:Curve:Quadratic 23

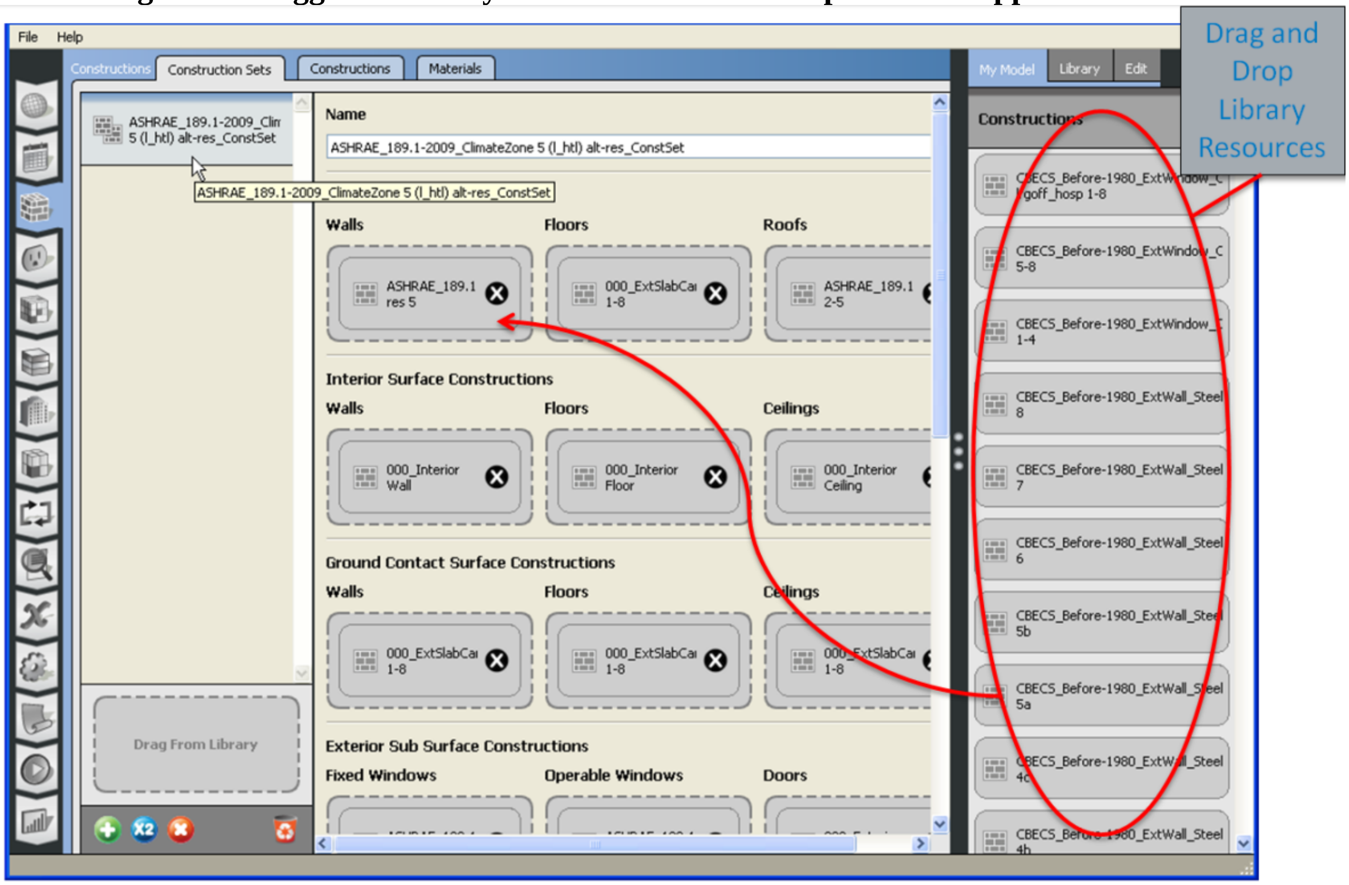
Energy Input Ratio Function of Temperature Curve Name
OS:Curve:Biquadratic 18

Credit: David Goldwasser/NREL

The singular OpenStudio Application (Figure 8) builds on and/or integrates the following example applications:

- The Plug-in for Google SketchUp – This plug-in was integrated into the new OpenStudio application and remains the "workhorse" of the suite for geometry visualization, space type management (and inherited constructions, schedules, loads, etc.), zoning, user extensible scripting, and simulation management.
- SystemOutliner – This application was introduced with OpenStudio 0.6, and is a visual interface to the SDK's mechanical system design capability, providing easy access to templated or custom HVAC system models.
- ResultsViewer – ResultsViewer was the first application built with the SDK, and serves as a means of viewing time series data from EnergyPlus.
- ModelEditor – This application provided a means of editing with OpenStudio's native .osm format, and was deprecated in OpenStudio 0.6 when sufficient editing functionality became available in the rest of the suite.
- RunManager – This is a highly flexible tool for simulation and scripted analysis management.

Figure 9. Draggable Library Resources within the OpenStudio Application



Name: ASHRAE_189.1-2009_ClimateZone 5 (l_htl) alt-res_ConstSet

Drag and Drop Library Resources
File
Help
Constructions
Construction Sets
Constructions
Materials
My Model
Library
Edit
Constructions
ASHRAE_189.1-2009_Clim 5 (l_htl) alt-res_ConstSet
ASHRAE_189.1-2009_ClimateZone 5 (l_htl) alt-res_ConstSet
Name
Walls
Floors
Roofs
ASHRAE_189.1 res 5
000_ExtSlabCar 1-8
ASHRAE_189.1 2-5
Interior Surface Constructions
Walls
Floors
Ceilings
000_Interior Wall
000_Interior Floor
000_Interior Ceiling
Ground Contact Surface Constructions
Walls
Floors
Ceilings
000_ExtSlabCar 1-8
000_ExtSlabCar 1-8
000_ExtSlabCar 1-8
Exterior Sub Surface Constructions
Fixed Windows
Operable Windows
Doors
Drag From Library
CBECS_Before-1980_ExtWindow_C ygoff_hosp 1-8
CBECS_Before-1980_ExtWindow_C 5-8
CBECS_Before-1980_ExtWindow_C 1-4
CBECS_Before-1980_ExtWall_Steel 8
CBECS_Before-1980_ExtWall_Steel 7
CBECS_Before-1980_ExtWall_Steel 6
CBECS_Before-1980_ExtWall_Steel 5b
CBECS_Before-1980_ExtWall_Steel 5a
CBECS_Before-1980_ExtWall_Steel 4c
CBECS_Before-1980_ExtWall_Steel 4b

Credit: David Goldwasser/NREL

As the OpenStudio SDK has become more capable, it has become increasingly straightforward to respond to user requests for new advanced and streamlined modeling processes. This is most evident in the deprecation of ModelEditor, and more recently, repurposing the SystemOutliner application to encompass a more comprehensive workflow.

A tabbed workflow leads the user through the complete process of building an energy model, including:

- Defining resources such as weather data, utility rates, constructions, schedules, loads, etc.,
- Applying resources to spaces,
- Zoning,
- Defining and applying mechanical systems to zones,
- Establishing simulation output variables,
- Specifying simulation parameters,
- Defining scripted user extensibility during model articulation, EnergyPlus input translation, and post-simulation,
- Running simulations,
- Visualizing high level results.

The BCL also figures prominently in the new application with the establishment of draggable resource libraries representing constructions, schedules, loads, etc. as shown in Figure 9. The near-term vision is for modelers to quickly identify data resources via integral multifaceted searching, and apply them visually to their models within the application.

Creating Custom Energy Analysis Applications Using the OpenStudio SDK

Four basic steps were repeated in each case study:

1. Define the problem
2. Choose the technology
3. Perform a gap analysis
4. Leverage the OpenStudio SDK

Defining the problem is key, sounds simple, and can be deceivingly challenging. The approach used for this paper's case studies used a collaborative group of subject matter experts, software developers, team leaders, interface designers, users, and stakeholders. The central tenet of the team was to adequately interview prospective users and determine their needs. "Must have" and "like to have" functionality was identified, and a prioritized requirements document was synthesized from the users interviews.

Choosing the technology is the next step. For the aforementioned case studies, robust freeware was the only software considered, and a rich collection is available. As an example, the Boost library is quite comprehensive, and Qt's library is rich in intrinsic functionality. The Ruby scripting language (along with its Rails extension) is broadly adopted, and a powerful means of extensibility for OpenStudio. In the realm of whole building energy modeling, NREL's OpenStudio SDK offers an easy entry point for unique application development, one that supports these robust technologies, and more.

A gap analysis can be performed once the problem is defined and the technologies have been chosen. This is where the strength of open-source truly shines; the ability to leverage and extend open-source software is invaluable when users develop their own applications. As the case studies seek to demonstrate, the NREL development team extends capabilities as needed to provide greater functionality with quarterly release cycle times.

To leverage the OpenStudio SDK, download a release or a developer package of OpenStudio from http://openstudio.nrel.gov/downloads. Developer releases are updated every 2 weeks, complete with Doxygen-generated documentation, and numerous Google GTests, which the development team uses as part of an automated, nightly quality assurance process. Major release packages are issued quarterly. An extensive collection of OpenStudio video tutorials is also available, with a greater emphasis on typical workflows and specific uses of the OpenStudio suite (http://www.youtube.com/user/NRELOpenStudio). A user forum at the OpenStudio website provides assistance to application users and developers alike (http://openstudio.nrel.gov/forum).

Conclusions

OpenStudio dramatically reduced the effort and risk required to develop the building energy modeling applications described in the four case studies. The OpenStudio SDK is continually evolving and becoming more feature-rich for building energy model articulation, transformation, and analysis, as well as domain-specific GUI development. OpenStudio uses extremely powerful open-source libraries, and because it is open-source, is highly extensible and customizable. The authors believe that the strength of open-source, collaborative development, and crowd-sourcing of supporting data via the BCL will create even greater opportunities for leverage.

Acknowledgements

This work was financially supported by the U.S. Department of Energy Building Technologies Program, and by the California Energy Commission. The authors are greatly indebted to our sponsors at those organizations, and also to our colleagues at the National Renewable Energy Laboratory. OpenStudio would not be what it is today without Brian Ball, Larry Brackney, David Goldwasser, Daniel Macumber, Luigi Gentile Polese, Alex Swindler, and Jason Turner. The 179D tax deduction case study work was largely done by Brent Griffith, Matt Leach, Eric Bonnema, and Michael Deru, in addition to a subset of the authors.

References

Crawley, D. B., J. W. Hand, M. Kummert, and B. T. Griffith, B.T. 2008. "Contrasting the Capabilities of Building Energy Performance Simulation Programs." *Building and Environment* 43(4): 661-673.

Deru, M., B. Griffith, M. Leach, E. Bonnema, R. Langner, and E. Hale. *179D DOE Calculator Technical Support Document*. National Renewable Energy Laboratory, Golden, Colo. To be published in late 2012.

Field, K., M. Deru, and D. Studer. 2011. "Using DOE Commercial Reference Buildings for Simulation Studies: Preprint." NREL Report No. CP-550-48588. National Renewable Energy Laboratory, Golden, Colo.

Guglielmetti, R., D. Macumber, and N. Long. 2011. "OpenStudio: An Open Source Integrated Analysis Platform." In *Proceedings of Building Simulation 2011*, Sydney, Australia.

Guglielmetti, R., S. Pless, and P. Torcellini. 2010. "On the Use of Integrated Daylighting and Energy Simulations to Drive the Design of a Large Net-Zero Energy Office Building." I *Proceedings of the Fourth National Conference of IBPSA-USA*, New York, N.Y.

Hale, E., and N. Long. 2010. "Enumerating a Diverse Set of Building Designs Using Discrete Optimization." In *Proceedings of SimBuild 2010*, New York, N.Y.

Hirsch, A., S. Pless, R., Guglielmetti, and P. A. Torcellini. 2011. "The Role of Modeling When Designing for Absolute Energy Use Intensity Requirements in a Design-Build Framework." NREL/CP-5500-49067, National Renewable Energy Laboratory, Golden, Colo.

Long, N., K. Fleming, and L. Brackney. 2011. “An Object-Oriented Database for Managing Building Modeling Components and Metadata.” In *Proceedings of Building Simulation 2011*, Sydney, Australia.

Pless, S. C. Lobato, J. Drexler, P. Torcellini, and R. Judkoff. 2011. “NREL’s Research Support Facility: An Energy Performance Update.” http://www.nrel.gov/sustainable_nrel/pdfs/rsf_operations.pdf.

Ward, G. J. 1994. The RADIANCE Lighting Simulation and Rendering System. Proc. SIGGRAPH.

Ellis, P. G., P. A. Torcellini, and D. b. Crawley. 2008. “Energy Design Plug-in: An EnergyPlus Plug-in for SketchUp.” In *Proceedings of the IBPSA-USA SimBuild Conference*, Berkeley, California.

Web Resources

Boost – http://www.boost.org
CLIPS – http://clipsrules.sourceforge.net
EnergyPlus – http://apps1.eere.energy.gov/buildings/energyplus
OpenStudio – http://openstudio.nrel.gov
Qt – http://qt.nokia.com
Ruby – http://ruby-lang.org
RubyGems – http://rubyforge.org/projects/rubygems
Ruby on Rails – http://rubyonrails.org
VTK Charts – http://www.vtk.org/Wiki/VTK/Charts
YouTube Videos for OpenStudio – http://www.youtube.com/user/NRELOpenStudio
179D DOE Calculator – http://apps1.eere.energy.gov/buildings/commercial_initiative/179d/