

# Enhancing User Customization through Novel Software Architecture for Utility-Scale Solar Siting Software

Brant Peery, R. Sam Alessi, Randy D. Lee, Leng Vang, Scott Brown, David Solan, Dan Ames

*Idaho National Laboratory, Idaho*

**Abstract:** The need exists for a spatial decision support application that allows users to create customized metrics for comparing proposed locations of a new solar installation. This document discusses how PVMapper was designed to overcome the customization problem through development of loosely coupled spatial and decision components in a JavaScript plug-in architecture, allowing the user to easily add functionality and data to the system. The paper also explains how PVMapper provides the user with a dynamic and customizable decision tool that enables them to visually modify the formulas that are used in the decision algorithms that convert data to comparable metrics. The technologies that make up the presentation and calculation software stack are outlined. This document also explains the architecture that allows the tool to grow through custom plug-ins created by the software users. Some discussion is given on the difficulties encountered while designing the system.

**Keywords:** photovoltaic, software, plug-in, JavaScript, customization, SunShot, U.S. Department of Energy.

## 1. SUNSHOT INITIATIVE AND PVMAPPER MISSION

Solar technologies have the potential to provide clean and secure energy to help meet the nation's power needs. Additionally, solar technologies provide an opportunity to further develop an emerging economic sector. Historically, solar energy has not made economic sense compared to other sources of energy due to the high costs associated with intermittency and photovoltaic (PV) production. Recently, technologies have seen improvements, specifically in the production of PV components, which has resulted in bringing down the costs and making solar a more viable option.

The U.S. Department of Energy's SunShot Initiative aims to achieve cost-competitive solar without subsidy by following a plan to realize a 75% overall cost reduction by the end of the decade. As PV module costs continue to drop, the non-hardware (i.e., soft) balance-of-system costs remain a large portion of the overall system cost. These soft balance-of-system costs include permitting, environmental requirements, and costs associated with social concern. The goal of the SunShot Initiative is for utility-scale solar developments within the continental United States to provide 15 to 18% of America's electricity. The PVMapper system will provide specific information to help developers select sites with minimal development costs and the greatest social acceptance.

The mission of PVMapper is to assist utility-scale solar development companies in making optimum siting decisions. PVMapper augments user knowledge and tools by providing the ability to do the following:

- To assess sites based on the quantifiable physical characteristics and the constraints of the natural resource (i.e., site properties)
- To provide and integrate a comprehensive set of site data and geographical layers

- To include measures for social risk and public acceptance
- To enable customization of site selection weighting and decision criteria (i.e., site scores).

This report discusses the general architecture for the site selection weighting and site comparison approach.

## 2. SOFTWARE APPROACH

PVMapper is a geographical information system (GIS) that provides users with the same type of information and capabilities that are common today with open source mapping tools. In addition to these mapping functions, PVMapper has added an independent site comparison capability. The site comparison framework is organized in such a way that information can flow between the PVMapper map data layer and comparison information layer.

### 2.1 Frameworks



Figure 1. Frameworks.

PVMapper utilizes Microsoft's open source MVC\_4.0 framework and is coded in C# on the server side. Much of PVMapper's functions run on the client's computer instead of a central server. Similar to most browser-centric applications, the client side uses JavaScript as its programming language. PVMapper utilizes several well-established frameworks in JavaScript in order to calculate and render the page to the client. Figure 1 shows the different frameworks and what part of the system uses them. Ext JS was selected as the overall page layout and display handler. Ext JS enables PVMapper to essentially become a single page application. PVMapper depends on the OpenLayers platform to process map data and display it on the screen. OpenLayers is used as a JavaScript GIS engine. While it is not fully featured when compared to server side systems, it is still very powerful and runs completely on the

client, which was a design goal of PVMapper. The GeoExt software is a great piece of community work that helps in the setup of OpenLayers objects into Sencha Ext JS controlled windows.

### 2.2 Modularity through Plug-ins

Separating the mapping function and decision-making function adheres to well-known software development patterns that require business logic to be organized separately from the data processing. This separation simplifies changing business rules (e.g., solar site selection rules) without disturbing code associated with processing map layers or calculating site properties, thus users can change their site selection preferences quickly and easily.

PVMapper's modules and module application programming interface provide the interface between map-based data processing and the PVMapper decision analysis structure. Modules are created as part of the initial development effort, but they also can be added at any time in the future, aiding the sustainability of the open source code. The modules are simply algorithms contained in JavaScript. The JavaScript code is written to conform to the plug-in architecture of the system following the definitions of the application programming interface. This process is simple enough that someone with very little JavaScript experience could successfully write a module that pulls data from a data source, displays a map on the application

map window, and calculates a score based on their data that would be automatically pulled into the scoreboard for each site.

Each custom module has one or more tool objects that will interact with the system. Currently, there are the following three types of tools (Figure 2):

1. ScoreTool, which will calculate scores for the scoreboard for each site.
2. InfoTool, which is used to add functionality to the system.
3. TotalTool, which is used to change the scoreboard summary statistics.

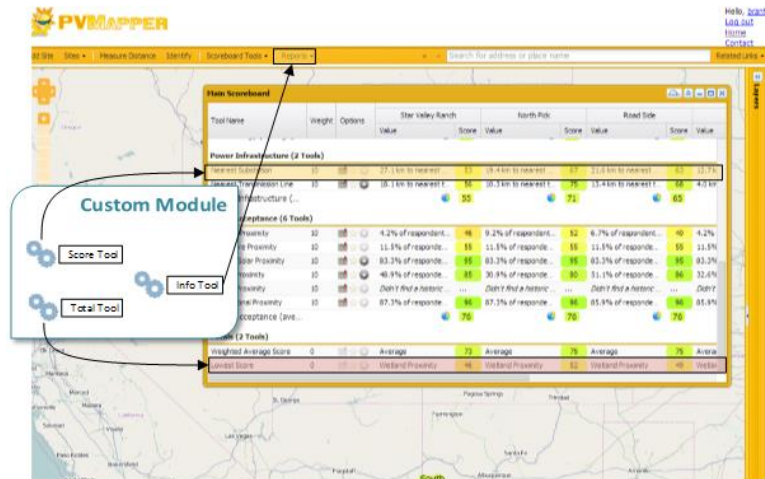


Figure 2. Tools uses.

Using one of these types of tools, a programmer can customize almost any part of the system without having to re-write base code. Many of the features in the current version of PVMapper are Idaho National Laboratory and Brigham Young University developed plug-in modules. Figure 3 provides an example of the minimal code that has to be written to enable a module that will score all sites that would be called “ThisTool” in the scoreboard.

```

1  var mytool: IScoreTool = {
2      title: "ThisTool",
3      description: "My super sweet score thingy",
4      init: null,
5      destroy: null,
6      activate: null,
7      deactivate: null,
8      updateScoreCallback: (score: pvMapper.Score) => {
9          score.updateValue(1);
10     },
11     onSiteChange: null,
12     onScoreAdded: function (context: any, event: EventArgs, score: pvMapper.Score) => void {}
13 };
14

```

Figure 3. Example module code (TypeScript).

By virtue of the module manager, the programmer does not have to worry about how the module will interact with the system. The programmer simply has to supply the needed properties and event handlers in the module code and the plug-in framework connects everything automatically. The controller for the PVMapper module handles recalculation of values for all user-created sites for each of the activated ScoreTools when needed. Using this technique, a module can be written that uses a unique data set, a unique map display, and unique data calculations. It could also provide derived values from its data in any way it needs to without the constraint of predesigned limits. This framework also allows the module to provide a configuration window for the user to further customize the tool to the user’s preferences. For example, the “Distance to a Power Line” tool uses a custom window to ask the user which type and rating of power line they would like to use to calculate distance. The tool provides the custom algorithm and the user provides the business logic without modifying code. This modular approach creates an environment where loosely coupled tools can be added to the system throughout the software’s life without having to change base code. This enhances the sustainability of the software, because it is easier to maintain and it can grow easily with new functionality. This open approach invites collaboration by the software users and ensures that PVMapper will stay relevant for the years to come.

## 2.3 Site Comparison through Multi-Objective Comparison

Solar site selection is a multi-objective decision analysis problem, because no single site property can be used to determine the best sites for solar development. Levels of solar insolation, access to roads and transmission lines, distance from sensitive wildlife habitat and residential areas, and so forth all influence site selection decisions. Many mathematical techniques (such as optimization, decision trees, simulation, and goal programming) of varying level of mathematical sophistication are available for comparing, contrasting, and ranking such decisions. Over time, PVMapper may incorporate multiple techniques; however, to begin, PVMapper has adopted a simple-to-understand, straightforward approach that has been used in many areas for decision making.

The approach, which is a type of multi-attribute utility theory<sup>1</sup> and is entitled the Scoring Function Approach, has been used for general engineering decision analysis (Wymore, 1993), automotive energy technology (Burns et al., 2004), natural resource ranking (Yakowitz et al, 1993), environmental quality, soil quality assessment (Karlen and Stott, 1994, Andrews and Carroll, 2001), and health care (Ruland, 2002). One educational website<sup>2</sup> states “One of the first applications of multi-attribute utility theory involved a study of alternative locations for a new airport in Mexico City in the early 1970s. The factors that were considered included cost, capacity, access time to the airport, safety, social disruption and noise pollution.”

The scoring function multi-objective approach is useful because it is easy for users to follow how numbers are transformed from site properties to dimensionless scores and weighted and rolled up into overall site comparisons. This numerical transparency facilitates interpretation and helps users to have more confidence in the comparisons produced.

This approach allows users to quantify their site preferences through the scoring function shape and location. Then, the scoring function maps the numeric property values that the user has deemed important for site comparison into a score value that lies between 0 and 1. Because all dimensional property values are mapped on a 0 to 1 dimensionless scale and are thereby normalized, properties with widely varying dimensional scales can be compared. The score is then multiplied by a weight assigned by the user and the product summed over all properties included in the overall site comparison scoring hierarchy.

$$Score(Site) = \sum_{x \in S} f(x) \text{ where } f(x) = (Weight(x) * ScoreFunction(Sitevalue(x))) \quad \text{Equation 1}$$

## 2.4 Scoring Functions

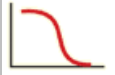




Table 1 shows example scoring function weightings and parameterizations. Simple increasing and decreasing curves are used, which are defined by the following six parameters: weight, scoring function, minimum value, target value, maximum value, and slope. The target value is the site parameter value for which the user prefers a middle score of 0.5. The minimum value is the smallest site value where the score reaches a limit of 0 or 1, depending on the scoring function form. The maximum value is the largest site value where the score reaches a limit of 0 or 1, depending on the scoring function form. Slope controls how rapidly or gradually the curve increases or decreases. The scoring function's shapes are often described by their name; for example, “Less is Better” is a curve transitioning from 1 to 0, “More is Better” is a curve transition from 0 to 1, “Center is Best” is a bell-shaped curve, and “Small and Large are Better” is an upside down bell-shaped curve. Scoring functions can take on many shapes<sup>3</sup> to represent stakeholder preferences.

<sup>1</sup> [http://wiki.ece.cmu.edu/ddl/index.php/Multiattribute\\_utility\\_theory](http://wiki.ece.cmu.edu/ddl/index.php/Multiattribute_utility_theory)

<sup>2</sup> [http://www.hsor.org/what\\_is\\_or.cfm?name=mutli-attribute\\_utility\\_theory](http://www.hsor.org/what_is_or.cfm?name=mutli-attribute_utility_theory)

<sup>3</sup> Many additional scoring functions can be used. Wymore (1993) lists 12 types, which include bell shapes, step shapes, and asymptotic forms.

Table 1. Example site selection preferences parameters and values.

Decision Category	Weight	Scoring Function	Scoring Function Parameters
<b>Environment: Road Access Distance</b>	40%	Less is Better 	Minimum: 0 miles Target: 2 miles Maximum: 20 miles Slope: -20
<b>Environment: Habitat Buffer</b>	60%	More is Better 	Minimum: 0.5 miles Target: 5 miles Maximum: 15 miles Slope: +20
<b>Energy: Net Annual Energy</b>	70%	More is Better 	Minimum: 30 MkW Target: 35 MkW Maximum: 40 MkW Slope: +20
<b>Energy: Intermittency</b>	30%	More is Better 	Minimum: 35% Target: 55% Maximum: 60% Slope: +20
<b>Social: Public Perception</b>	100%	More is Better 	Minimum: 80% Target: 98% Maximum: 100% Slope: +20

The scoring functions are designed to take a value that is produced by a module tool and translate that value into a dimensionless and normalized value between 0 and 1. The functions in Table 1 show how this could be accomplished when the tool provides numeric values. In order to normalize textual values, PVMapper uses a user-configurable star rating system illustrated in Figure 4. The default ratings are provided by the module, giving the developer of the tool a way to present sane defaults. The user can reconfigure the ratings to match their preferences, which will affect the value that is used in the selected score function. In this way, a textual value can be transformed into a numeric value that then is normalized through the same method as the other tools.

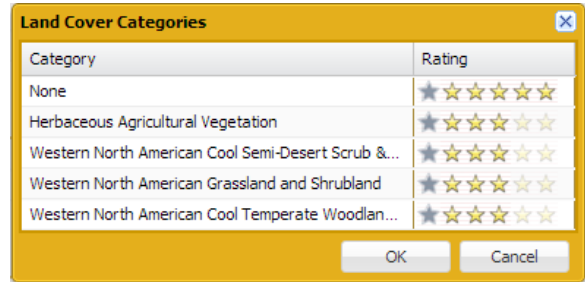


Figure 4. Example star rating.

## 2.5 Scoring Calculation Plug-in Architecture

The available scoring functions in the function collection can be modified by virtue of PVMapper's plug-in architecture (Figure 5). Using the native strengths of JavaScript to enhance an object through object injection, an author could create an InfoTool plug-in that adds or overrides a function to the system. All object and properties of the utility function system can be overridden by independently created code. By using loosely coupled utility function code, the software is able to morph and grow to fit the exact needs of the user.

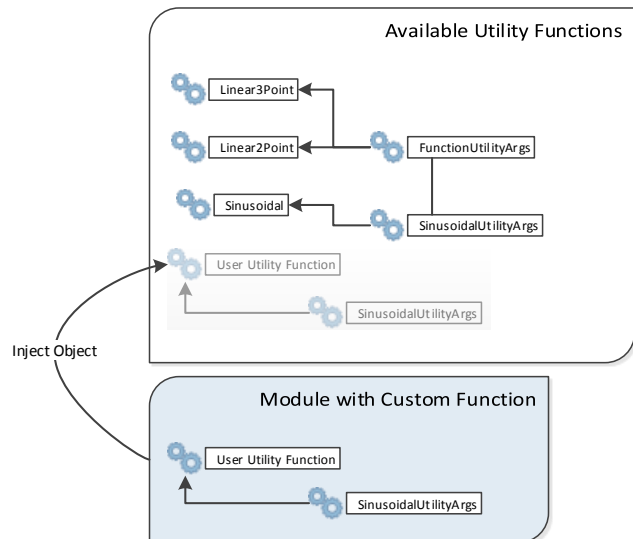


Figure 5. Modular utility functions.

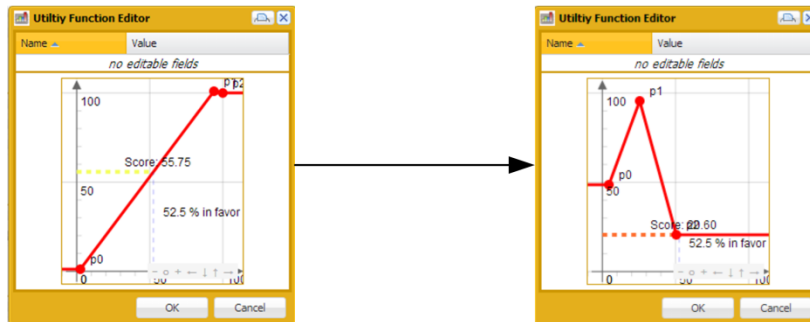


Figure 6. Modifying a function.

The utility functions also are changeable by the user without the need for code. The calculations are run in a parameterized function, which allows the user to modify the parameters to more closely match their desired scenario. For example, if a user wanted a “more is better”-type function to become a “more is better up to a point” (say 30) and then the desirability of the site drops off quickly to 50 but is always somewhat desirable, a function could be modified by the user from the default (see Figure 6).

### 3. SUMMARY

PVMapper is designed to help utility-scale solar development companies make siting decisions. In addition to a basic GIS capability, PVMapper has added an independent site comparison analysis framework. The two components interact to provide users with information on the properties of the sites, as well as scores that represent a convolution of site properties and user-oriented site comparison preferences. These site comparison preferences can consist of a combination of predefined regulatory-oriented and user-oriented preference choices.

In the current and initial version of PVMapper, scores are calculated using the scoring function approach. In future versions of PVMapper, additional decision analysis tools could be added. The scoring function approach is attractive because of its simplicity and easy user interpretability. An overview of the modular plug-in architecture is given in regard to the user interface, map data aggregation and value extraction, and utility scoring functions and their respective graphical user interfaces. The modular approach creates an excellent opportunity for PVMapper to grow with the user’s needs. The architecture is open, thus it is more sustainable into the future. It has been shown how PVMapper is a highly customizable software designed to meet the broad needs of the various users who will be interacting with the software.

### 4. AUTHOR CONTACT INFORMATION

Brant Peery, Idaho National Laboratory, Idaho Falls, Idaho; brant.peery@inl.gov  
Sam Alessi, (Recently retired from Idaho National Laboratory); sam@almanac-systems.com  
Randy Lee, Idaho National Laboratory, Idaho Falls, Idaho; randy.lee@inl.gov  
Leng Vang, Idaho National Laboratory, Idaho Falls, Idaho; leng.vang@inl.gov  
Scott Brown, Idaho National Laboratory, Idaho Falls, Idaho; s.brown@inl.gov  
Dan Ames, Brigham Young University, Provo, Utah; dan.ames@byu.edu  
David Solan, Boise State University, Boise, Idaho; davidolan@boisestate.edu

### 5. REFERENCES

Andrews, S.S., Carroll, C.R., 2001. Designing a decision tool for sustainable agroecosystem management: Soil quality assessment of a poultry litter management case study. *Ecol. Applic.* 11 (6), 1573–1585.

- Burns, J.W., Cors, T., Knight, B., Thelen, B., 2004. Evaluating Advanced Automotive Energy Technologies: A Multivariate Contribution Metric. *Intl. J. of Energy Technol. and Policy* 2(3), 262–271.
- Karlen, D.L., Stott, D.E., 1994. A framework for evaluating physical and chemical indicators of soil quality. In: Doran, J.W., Coleman, D.C., Bezdicek, D.F., Stewart, B.A. (Eds.), *Defining Soil Quality for a Sustainable Environment*. SSSA, Inc., Madison, Wisconsin, USA.
- Ruland, C., 2002. Patient Preferences in Health Care Decision Making.  
<http://people.dbmi.columbia.edu/~cmr7001/sdm/html/definitions.htm#multi>.
- Wymore, W., 1993. *Model-Based Systems Engineering*, CRC Press, Boca Raton.
- Yakowitz, D.S., Stone, J.J., Lane, L.J., Heilman, P., Masterson, J., Abolt, J., Imam, B., 1993. A decision support system for evaluating the effects of alternative farm management systems on water quality and economics. *Water Sci. Technol.* 28 (3-5), 47-54.