



**CIFE** CENTER FOR INTEGRATED FACILITY ENGINEERING

CAD-Centric Attribution  
Methodology for Multidisciplinary  
Optimization (CAMMO):  
*Enabling Designers to Efficiently  
Formulate and Evaluate Large  
Design Spaces*

By

**Benjamin Welle, John Haymaker,  
Martin Fischer, & Vladimir Bazjanac**

**CIFE Technical Report #TR195  
January 2011, revised March 2012**

**STANFORD UNIVERSITY**

**COPYRIGHT © 2012 BY**  
**Center for Integrated Facility Engineering**

If you would like to contact the authors, please write to:

*c/o CIFE, Civil and Environmental Engineering Dept.,  
Stanford University  
The Jerry Yang & Akiko Yamazaki Environment & Energy Building  
473 Via Ortega, Room 292, Mail Code: 4020  
Stanford, CA 94305-4020*

Journal Submission to Advanced Engineering Informatics:

**CAD-Centric Attribution Methodology for Multidisciplinary Optimization (CAMMO):**  
*Enabling Designers to Efficiently Formulate and Evaluate Large Design Spaces*

Benjamin Welle <sup>a,1</sup>, John Haymaker <sup>b,2</sup>, Martin Fischer <sup>a,3</sup>, Vladimir Bazjanac <sup>c,4</sup>

<sup>a</sup> Center for Integrated Facility Engineering (CIFE), Department of Civil and Environmental Engineering, Stanford University, Stanford, CA

<sup>b</sup> Design Process Innovation, San Francisco, CA

<sup>c</sup> Building Technologies Department, Environmental Energy Technologies Division, Lawrence Berkeley National Laboratory (LBNL)

<sup>1</sup> Environment and Energy Building, 473 Via Ortega, #291, MC:4020, Stanford, CA 94305, Phone: (650)-387-9925, Email: [bwelle@stanford.edu](mailto:bwelle@stanford.edu)

<sup>2</sup> 477 Vermont St., San Francisco, CA 94107, Phone: (415)-290-2738, Email: [johnrhaymaker@gmail.com](mailto:johnrhaymaker@gmail.com)

<sup>3</sup> Environment and Energy Building, 473 Via Ortega, #297, MC:4020, Stanford, CA 94305, Phone: (650)-725-4649, Email: [fischer@stanford.edu](mailto:fischer@stanford.edu)

<sup>4</sup> Lawrence Berkeley National Laboratory, University of California, MS 90-3111, Berkeley, CA 94720, Phone: (510)-486-4092, Email: [v\\_bazjanac@lbl.gov](mailto:v_bazjanac@lbl.gov)

## ABSTRACT

Research has demonstrated that multidisciplinary design optimization (MDO) processes that automate the workflow from a parametric product model to performance simulation engines can compress design cycle time, increase design knowledge, and yield substantive product quality and performance gains. However, the accuracy and cost-effectiveness of an MDO process is highly dependent on designers' ability to structure the optimization problem for specific challenges, particularly when specifying how building attributes and their associated geometry are configured for an optimization process. To fit current workflows efficiently, designers need flexible CAD-centric attribution methods for MDO environments. These methods are not addressed in literature, or defined in available methods. This paper fills these gaps by developing a CAD-Centric Attribution Methodology for Multidisciplinary Optimization (CAMMO). The authors demonstrate the potential power and generality of CAMMO with two industry case studies.

**KEYWORDS:** multidisciplinary design optimization (MDO); conceptual building design; CAD-centric attribution; energy simulation; daylighting simulation; process integration and design automation

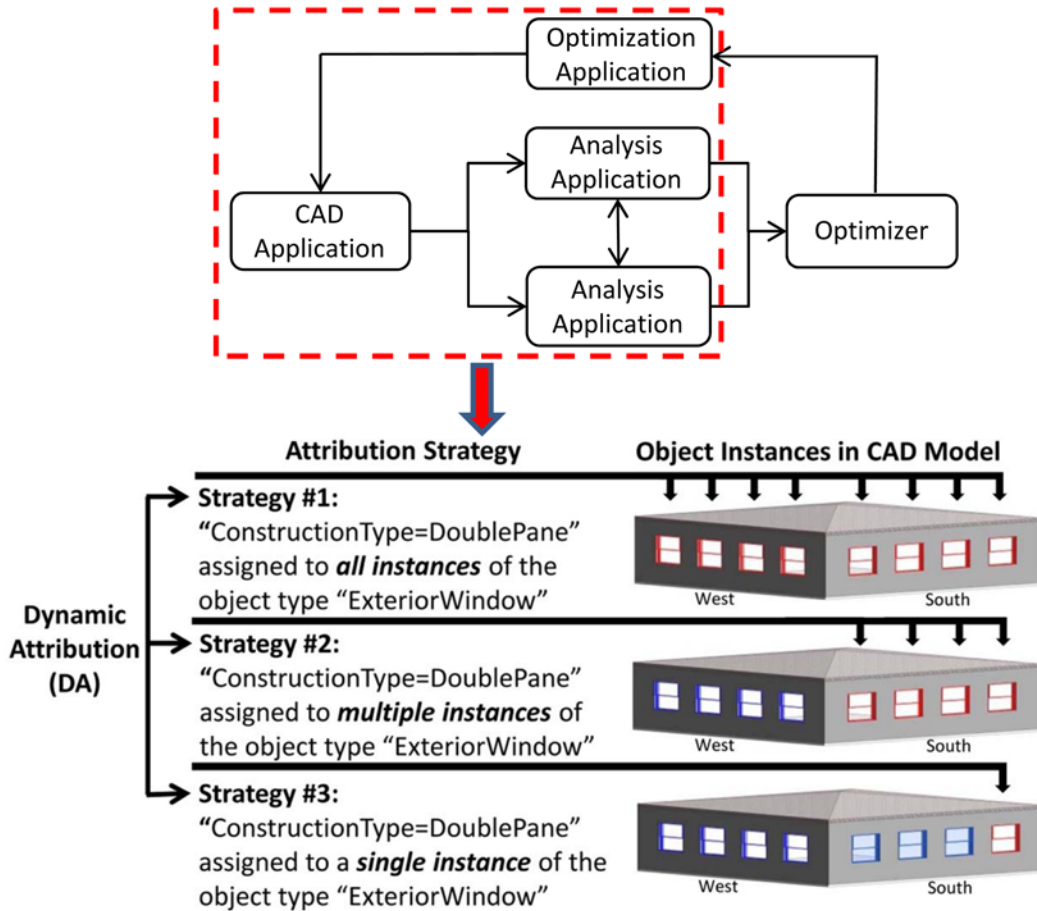
## 1. INTRODUCTION

The greatest opportunity to reduce energy consumption in buildings lies in the concept design phase, when the designer defines orientation, massing, materials, components, and systems. However, a number of tool and process limitations currently hinder the ability of designers to adequately explore and evaluate the design spaces they would like to explore [1-4]. To demonstrate the magnitude of the design spaces facing conceptual building designers, consider the simple example of a rectangular building with 4 windows on both the south- and west-facing walls, and 10 possible windows construction types. A designer may want to ask: "what is the impact of independently modifying the construction type for:

	<u>Possible Alternatives</u>
Scenario 1: all the windows?"	10
Scenario 2: the windows on each orientation?"	100
Scenario 3: each individual window?"	100,000,000

Each of these three questions represents three different problem formulations with their own vast alternative and impact spaces that designers cannot adequately explore using conventional methods. Multidisciplinary design optimization, or MDO, is one possible design method to enable faster, more reliable, and more systematic evaluation of alternatives, which research suggests increases the probability of discovering higher performance designs [5, 6].

During a recent investigation of the potential of MDO for AEC, the authors implemented a simple classroom case study evaluating it for structural integrity, energy consumption, daylighting, and initial capital and life-cycle costs [7]. However, the automation process implemented lacked flexibility for it was only possible to modify the construction type for all the exterior windows at the same time (Scenario 1 above). Enhancements to the automated process to support Scenarios 2 and 3 would have been time consuming and costly, suggesting that the accuracy and cost-effectiveness of MDO is highly dependent on the ability of the selected automation process to support a diverse set of problem formulations [8]. Designers require a robust method for associating attributes needed for downstream applications with parametric, object-based CAD geometry, or *CAD-centric attribution* [9]. This functionality would enable the flexibility in problem formulation shown in the three attribution strategies in Figure 1. The authors refer to a method that supports all these attribution strategies simultaneously without the need for further software development as *dynamic attribution (DA)*.



**Figure 1:** A general CAD-based MDO process consists of a CAD application (with GUI), multiple analysis applications (GUI optional), an optimization application (with GUI), and an optimizer. Dynamic attribution is a method that allows the assignment of a given attribute value to different groupings of object instances in a CAD model upstream of the analysis applications. In this example, Strategy #1 assigns the attribute value “ConstructionType=DoublePane” to all *all instances* of the object type “ExteriorWindow”. Strategy #2 assigns the same attribute value to all the *multiple instances* of the object type “ExteriorWindow”, while strategy #3 assigns it to one *single instance* of the object type “ExteriorWindow”. Key: Blue=SinglePane, Red=DoublePane.

Such a method is critical for the successful application of MDO and automated building simulation environments to the simulation of building passive thermal performance, which refers to all non-mechanical energy flows in a building, due to the complexity of today’s sustainable design objectives and the resulting wide range of analysis input configurations possible during the conceptual design phase. Passive thermal design is the evaluation of the thermal performance of passive thermal design strategies during the building design process. Literature asserts that passive thermal design is the most critical component of net-zero energy (NZE) design, given that the steps towards a NZE design, in order of importance, are to (1) reduce thermal loads; (2) meet thermal loads with most efficient HVAC equipment possible; and (3) meet the remaining demand with renewable energy [10-15].

The following sections review previous research in flexible problem formulation for MDO, CAD-centric MDO, and CAD-centric attribution for non-MDO and MDO environments, culminating in the authors’ proposed research contribution with this paper.

### **1.1. Flexible Problem Formulation for MDO**

Prior research has identified flexible problem formulation as critical for successful application of MDO [16-18]. However, literature defines problem formulation in the context of MDO as pertaining primarily to the automation process *downstream* of executing the analysis applications [19, 20]. For example, literature identifies the need for flexibility in problem formulation construction in terms of the ability to employ various optimization algorithms (e.g., GA vs. gradient-based) [21], enabling the designer to configure complex process branching through a visual programming interface [22], or for connecting, replacing, deleting, and adding process nodes to the problem [20]. Another major area of emphasis is in the sequencing of optimization algorithms, including Collaborative Optimization (CO) to implement a system-level optimizer in conjunction with discipline-specific optimizers [23], Interactive Collaborative Optimization (ICO) for the inclusion of user interaction in the process [24], Concurrent Sub-Space Optimization (CSSO) [25], and Bi-Level Integrated System Synthesis (BLISS) [26]. However, the limitations in problem formulation capabilities encountered in the case study were due to decisions made *upstream* of the analysis applications, where designers specify geometry and other required inputs for analysis (Figure 1). No MDO literature addresses this component of problem formulation, nor does AEC literature for semi-automated building thermal simulation environments [18, 21, 27-35].

### **1.2. CAD-Centric MDO**

Performance-based design supported by product models is becoming state-of-the-art practice for an expanding number of design disciplines [36]. Product models are now being applied to all phases of the design process, from the generation of design concepts through preliminary and detailed design. The use of these models, also called building information models (BIM) [37], allows practitioners to efficiently generate and modify geometric and semantic models. There is widespread support for a product model centric approach to MDO in literature [2, 21, 38, 39].

### **1.3. CAD-Centric Attribution**

The automation process upstream of analysis applications for CAD-centric MDO includes the coupling of CAD and Computer-Aided Engineering (CAE) through geometry-based analysis attribute specification [40, 41]. For this research, the terms CAD and product model are used interchangeably and assumed to both employ an object-oriented data structure, which is sometimes referred to as feature-based modeling [42, 43]. CAD-centric attribution is a fundamental and necessary part of an object-based problem definition and the most efficient method of prescribing this information due to the ease of access, re-usability, and management of object-attribute relationships throughout the product lifecycle [44, 45]. Currently, designers typically apply analysis attributes after the model has been abstracted and prepared for analysis. These analysis attributes are not permanently associated with the CAD geometry and may require recreation each time the CAD geometry changes. Pre-processing for highly complex CAE analyses can sometimes require weeks of effort. While attribution within the CAD GUI has become more frequent in the AEC industry, the process is ad hoc, is limited to a single attribute value per attribute, and typically has no formal method for how the attributes and their values are stored and managed during a multidisciplinary design and analysis process.

CAD-centric attribution consists of two steps: (1) the assignment of attributes to CAD geometry and (2) the assignment of values to those attributes. An attribute management system (AMS) is a tool to assist the designer in applying, deleting, and editing attributes to a CAD model and maintaining consistency in the analysis models. An object-oriented data structure and an intuitive graphical user interface are needed to create an AMS [41, 46]. When applying attributes to CAD geometry, it is important to determine the best method to associate, store, and manage them. Attribute assignment could take place in the AMS GUI or directly in the CAD GUI. Attribute and geometry storage after attribution could take place within the

CAD database, in an external database, or via file input/output (I/O) using a custom or neutral file format (e.g. STEP or IGES). An AMS could also include methods for the transfer/conversion of attributed geometry to one or more analysis applications. This AMS component could use file I/O or direct translators [47], as well as one or more pre-processors/middleware and analysis application GUIs.

Past research has investigated a wide range of AMS methods. King developed a custom GUI to access the CAD model via the application programming interface (API), list the available parts in the product model, and allow the designer to assign sets of attributes to selected part objects or face entities. The geometry-attribute relationships were then stored back in the native CAD geometry database [46]. Arabshahi developed a similar AMS that stored the attributed geometry in an external database for translation into the analytical model via direct translators [45]. Baker developed an attribution method using database management systems (DBMS), similar to centralized model servers [48], for defining, instantiating, and managing attributes in the CAD environment, with CAE applications retrieving the geometry via the CAD API and associated attributes from the DBMS via attribute translators [49]. O'Bara implemented an object-oriented, CAD-centric attribute management system with attributes stored or retrieved in terms of file I/O [50]. Although some of these CAD-centric attribution methods suggest the need for flexibility in assigning attributes and attribute values to different groupings of CAD objects, none of them specifically address CAD-centric attribution for MDO environments [44, 49, 51, 52] and each would inadequately support dynamic attribution for fully-automated modeling and simulation environments. The authors discuss the specific considerations for CAD-centric attribution in MDO environments in the following section.

#### **1.4. CAD-Centric Attribution for MDO**

CAD-centric attribution for MDO requires a unique set of considerations when choosing how to structure the AMS and overall automation process, and is dependent on the desired MDO environment. An increasingly popular MDO environment called Process Integration and Design Optimization (PIDO) helps engineers and analysts manage the setup and execution of simulation and analysis tools, integrate and synthesize results from multiple domain applications, and optimize one or more aspects of a product design [19, 20, 38, 39]. PIDO environments enable the integration of commercial or proprietary software tools into a common environment using software “wrappers” or “plugins” which interface with the tools to be automated. Given the distributed nature of design and analysis tools in the AEC industry, this type of MDO environment is likely to be an important option for design firms that decide to integrate MDO into their design workflows. Consequently, when considering a PIDO environment, the developer must address several important requirements in the development of a CAD-centric attribution methodology:

1. ***CAD GUI attribution***- Requiring the designer to assign attributes directly to CAD objects in the CAD GUI has benefits and drawbacks. Requiring optimization loops to go back through the CAD model simply to change a non-geometric attribute is inefficient and more unstable than a shorter inner loop that excludes the CAD model. The degree of attribution granularity desired, potential tradeoffs in time and complexity when compared to a greater degree of attribution in the AMS GUI, and the chosen method to expose variables to the Optimization GUI will all impact the value of direct attribution in the CAD GUI. The developer must specify clear rationale for requiring attribution of the geometry model within the CAD GUI.
2. ***Geometry model file export***- It is common for analysis applications (or their pre-processors) to require CAD topology data generated only through a geometry file export from the CAD tool. Using a neutral file format to transfer geometry may be desirable if it enables interoperability with multiple desired analysis applications or implements a required geometric topology generation process. The geometry model may or may not be attributed based on the previous requirement.

3. ***Automation of non-attribution CAE pre-processing-*** Since optimization algorithms may need to analyze thousands of design iterations before finding an optimal solution, it is critical to automate data transformations, reductions and simplifications, translations, and/or interpretations [53] to convert the architectural model to an analytical model. This step is of particular importance when automating the workflow of multiple analysis applications that have different geometry models. The location of the required pre-processor impacts the structure of an attribution method.
4. ***Specification, storage, and management of attributed geometry model-*** The designer requires methods to add/delete/edit attributes and/or modify attribute values, as well as to store and manage them. A robust method would also include mechanisms to deal with missing or incorrect information in the CAD application [54]. Places the designer could do this include within the CAD GUI/database, file input/output (I/O), or an external database management system (DBMS). Each method has their strengths and weaknesses. For example, access to a CAD database may be limited and slow, though it performs more efficiently than potentially time-consuming and inconsistent file I/O. Though a DBMS may add considerable development requirements, it could also provide improved functionality with the ability to specify additional attributes via an AMS with multiple GUIs and serve as a central repository of data accessible by a wide range of applications. The selection of attributes to expose and their corresponding structure is a critical decision if an AMS GUI/DBMS combination is desired, particularly when considering its use in an MDO/PIDO environment. This decision is important because the automated process must eventually expose selected attributes as optimization parameters in the Optimization GUI, resulting in a key departure from past AMS methods.
5. ***Transfer of attributed geometry to multiple analysis applications-*** Analysis wrappers must access and utilize the attributed (and possibly abstracted) geometry model, whether it is directly from the CAD database/API, via an external database, or by some other means.
6. ***Dynamic attribution for optimization and non-optimization parameters-*** Designers require methods to dynamically associate and specify ranges of attributes to different groupings of object instances in the product model, as described in Section 1. The method for enabling dynamic attribution directly impacts, or is impacted by, the previous five considerations.

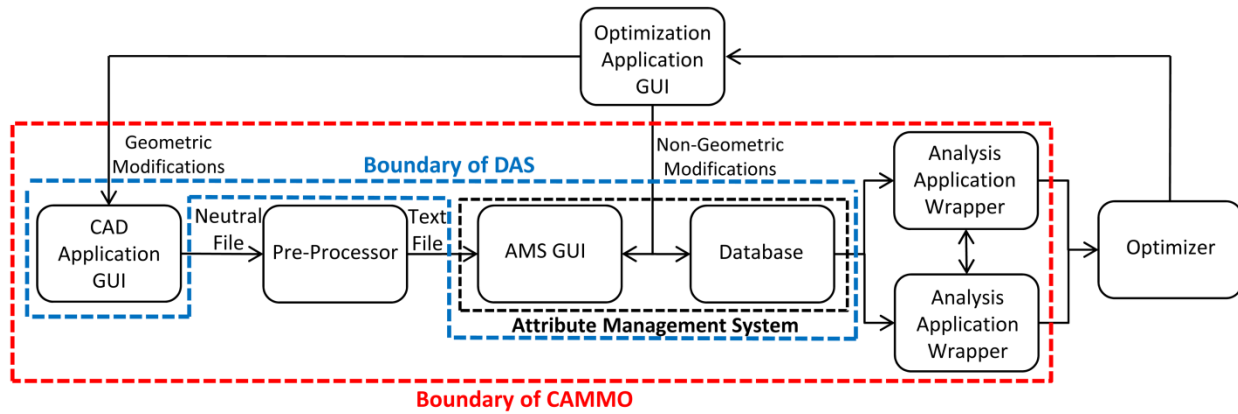
Currently no literature adequately addresses these considerations in a holistic manner for CAD-centric MDO attribution. To fill these research gaps, this paper develops, implements, and provides evidence for the power and generality of a dynamic attribution method called the CAD-Centric Attribution Methodology for Multidisciplinary Optimization (CAMMO). The authors apply this methodology to the evaluation of passive thermal design strategies using annual energy and daylighting simulation tools. With this methodology and some additional resource investment during the initial automation process to enable it, optimization accuracy will improve and overall costs in the application of MDO to support multiple problem formulations is minimized.

## **2. CAD-Centric Attribution Methodology for Multidisciplinary Optimization (CAMMO)**

This section summarizes the development of CAMMO, beginning with the definition of the product model concepts used in this research. The section then describes the major components of CAMMO, including: (1) a CAD application GUI/wrapper and corresponding user requirements; (2) a geometry file export using a neutral file format; (3) a pre-processor; (4) an AMS with an external database; (5) analysis application wrappers; and (6) a method for dynamic attribution called a dynamic attribution structure (DAS). Each of these components corresponds, in sequence, to the CAD-centric attribution requirements in Section 1.4. Figure 2 shows a high-level overview of the workflow for the first five components, as

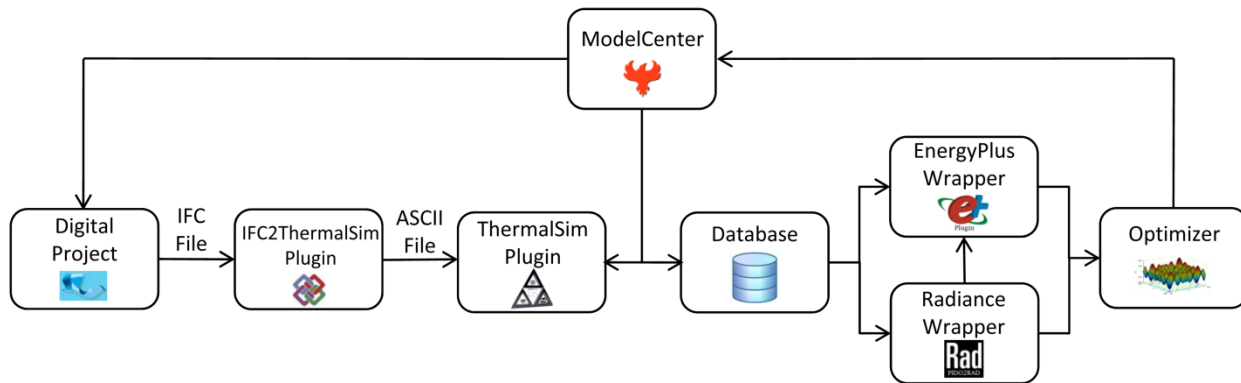


well as the boundary of the DAS. Figure 3 shows the authors' technical implementation of the workflow. Details of the DAS are provided in Figure 4-Figure 6. The authors developed CAMMO specifically for PIDO environments.



**Figure 2:** CAMMO consists of a CAD Application GUI (CAD GUI), a neutral file format CAD export, a pre-processor with a text file export, an AMS consisting of a GUI and an external database, analysis application wrappers, and a DAS that utilizes the CAD GUI and AMS. The boundary of the DAS is in blue. The red line shows the boundary of CAMMO in the context of the overall optimization workflow.

Figure 3 shows the authors' technical implementation of CAMMO. The MDO framework selected is ModelCenter [55], the parametric CAD application is the Architecture and Structures (A&S) Workbench in Digital Project [56], the energy simulation engine is EnergyPlus [57], and the daylighting simulation engine is Radiance [58]. An IFC file passes information from Digital Project downstream. This implementation is described in more detail in previous research [59].



**Figure 3:** The designer starts the MDO process by building a parametric CAD model and assigning analysis information in Digital Project. The attributed model is exported via IFC to a pre-processor (IFC2ThermalSim Plugin) then via an ASCII file to the AMS GUI (ThermalSim Plugin) where the designer assigns some additional information such as optimization/simulation settings and additional analysis attributes. The resulting data is stored in an external database and accessed directly by the EnergyPlus and Radiance Wrappers to create input files for simulation.

## 2.1. Product Model Concepts

The authors must define an attribute taxonomy before implementing an attribution methodology. Various taxonomies for attributes have been proposed by research [9, 46, 51, 60]. This research uses a class-based

attribute taxonomy [51]. Class 1 attributes are global, meaning required for analysis but independent of geometry. Class 2 attributes are associative and simple, meaning attached to geometry but independent of other geometric elements/attributes. Class 3 attributes are associative and complex, meaning attached to geometry and dependent on other geometric elements/attributes. If the complex attribute is dependent on other geometry or non-geometric attributes, it is considered geometric or non-geometric, respectively.

Both research and CAD vendors provide a wide range of heterogeneous definitions for various product model concepts. Table 1 clarifies the primary product model concepts for this research [37, 61], as well as several secondary product model concepts that are used for convenience.

<b>Product Model Concept</b>	<b>Description</b>	<b>Examples</b>
<i>Primary</i>		
Object Family	A pre-defined set of base objects in a product model.	Window
Object Instance	A unique object in a product model by Object Family.	Window.1
Object Type	A subtype of a given Object Family based on building function and/or behavior.	ExteriorWindow
Object Attribute	A property of an Object Instance required for analysis.	ConstructionType
Object Attribute Value	The value of the Object Attribute.	DoublePane
Object Group	An Object Attribute that identifies similar Instances of a given Object Type based on user-defined criteria.	Plan South
<i>Secondary</i>		
Object Type Instance	A unique object of a given Object Type (i.e., a subset of Object Instances).	Window.1
Object Type Attribute	A property of an Object Type Instance (i.e., a subset of the Object Attributes).	ConstructionType
Object Type Attribute Value	The value of the Object Type Attribute.	DoublePane

**Table 1:** CAMMO builds upon six primary and three secondary product model concepts: Object Family, Object Instance, Object Type, Object Attribute, Object Attribute Value, Object Group, Object Type Instance, Object Type Attribute, and Object Type Attribute Value (not capitalized for remainder of the paper).

Object instances are grouped by object type rather than object family because object function and/or behavior in the building model drives performance in analysis applications, and therefore is the most appropriate level by which to associate attributes to geometry for analysis [37]. Object families for this research are wall, window, door, slab, column, space, and shading objects. Object types for each object family are (1) wall: exterior wall and interior wall; (2) window: exterior window and interior window; (3) door: exterior glass door, interior glass door, exterior door, and interior door; (4) slab: base slab, floor slab, roof slab, ceiling, and floor; (5) column: none; (6) space: private office, open office, hallway, conference room, etc.; and (7) shading: overhang, fin, outer lightshelf, inner lightshelf, and adjacent building.

## 2.2. CAD Application GUI/Wrapper

The designer builds a parametric model in the CAD Application GUI, and exposes the driving parameters for geometry as optimization parameters in the MDO environment via the CAD API/wrapper. The designer assigns associative attributes (Class 2 and Class 3) and initial attribute values in the CAD GUI through the assignment of custom property sets (p-sets) to all the object instances in the model. Next, the designer selects values for each of the applicable attributes from either a drop-down menu linked to a list

of available attribute enumeration values or by entering a string value. The rationale for initial assignment of non-geometric attributes using the CAD GUI rather than the AMS is to (1) allow designers to see objects within the 3D building context before selecting attribute values; (2) minimize the attribution time requirements in the AMS; and (3) prevent problematic exposure of attributes for optimization purposes in the AMS and Optimization GUIs due to an unmanageable number of design variables that would have to be exposed for optimization. One potential drawback of relying solely on assigning object type attribute values in the CAD model is the need to loop back through the CAD model when modifying object type attribute values during optimization, resulting in excessive computational overhead for non-geometric variations. CAMMO prevents this inefficient workflow by having modifications to object type values made in the AMS external database directly. Section 2.7 further discusses this last step.

### **2.3. Geometry Model File Export**

Past research in CAD-centric attribution has suggested that the use of geometry file exports to transfer geometry to product data management systems is inefficient and less desirable than accessing the CAD system's geometric database directly due to the degradation or loss of the parametrics governing the rules and relationships in the model [46, 49, 62]. However, currently several file formats exist that use an object-based data model specific to their domain that do not suffer the type of data degradation experienced with past geometry file formats. The authors have chosen to transfer geometric and non-geometric data from the CAD model downstream for further processing using the neutral file format Industry Foundation Classes (IFC) [63]. Upon export of the IFC file, the CAD application's exporter automatically generates a wide range of topological data pertaining to object relationships and other data required for downstream applications. While some of the topological data may be accessible via the API with some extensive programming, some data generated by the export is inaccessible via the API. Another reason for using IFC is the potential for compatibility with a wide range of preprocessing tools and analysis applications. The optimization platform makes modifications to model geometry directly in the CAD geometry database via the CAD API.

### **2.4. Pre-Processor**

Geometric and non-geometric data from the CAD model may need to be pre-processed to meet the needs of downstream analysis applications. In traditional CAD to analysis I/O workflows, this pre-processing step takes place after the export of the geometry/neutral file format and before import into the analysis application's internal database. However, when including an AMS in the workflow, the developer must decide whether to put the pre-processor before or after the AMS. This decision is based on the function of the pre-processor. If the outputs of the pre-processor are required for the AMS to function correctly, then the developer must place the pre-processor before it. If not, the developer may place the pre-processor after the AMS. For example, the purpose of the pre-processor in the authors' workflow [59] is to calculate thermal space boundaries from architectural space boundaries, convert column objects into shading surfaces, split window objects with lightshelves, re-position overhangs and fins for non-planar walls and windows, and post-process object p-sets from the IFC file. The outputs of this pre-processor are required before the attribution process can take place within the AMS. Consequently, the authors placed the pre-processor before the AMS.

### **2.5. Attribute Management System**

Research has suggested that a DBMS provides greater efficiency, flexibility, and functionality than other data storage and management methods through the use of a centralized repository that a wide range of downstream applications may access, possibly on a server that multiple users can access over a network [46, 49, 64, 65]. The DBMS option is of particular relevance for this research, for the storage mechanism

must reside outside of the CAD environment due to dependency of the AMS GUI functionality on the pre-processor output, which is already outside of the CAD environment.

CAMMO utilizes an AMS consisting of a GUI and an external database. The GUI serves as the attribution interface for multiple analysis applications, reducing user input requirements. Upon launching the GUI, the designer imports the text file generated by the pre-processor. The GUI scans the file, exposes only the attributes for object types present in the text file, and then stores the contents of the text file in the database. The database contains libraries modifiable via excel spreadsheets that contain valid enumerations for attribute values, including default values for both associative and global attributes. The GUI accesses these libraries and exposes the valid enumerations to the designer via pull-down menus for each attribute. Once the designer selects the global and associative attribute values, the information is stored back in the database for use by downstream analysis application wrappers. The designer defines optimization parameters in the Optimization GUI. This component architecture allows the Optimization GUI to make any non-geometric attribute changes directly in the DBMS, eliminating the need to access the CAD model again, reducing time and complexity.

The AMS GUI exposes a pre-defined list of attributes for each object type in the model. Attribute definitions cannot be added, deleted, or modified by the designer in the GUI. This limitation is justified for several reasons. First, input requirements of the analysis applications and the optimization capabilities desired by the designer determine the specification of which attributes to assign a given geometric entity, both of which are typically known before the automation process takes place. Second, if the designer desires to add new attribute definitions for a given object type, modifications are required to the CAD GUI and AMS GUI to expose them and the analysis application wrappers to recognize and correctly process them.

## **2.6. Analysis Application Wrappers**

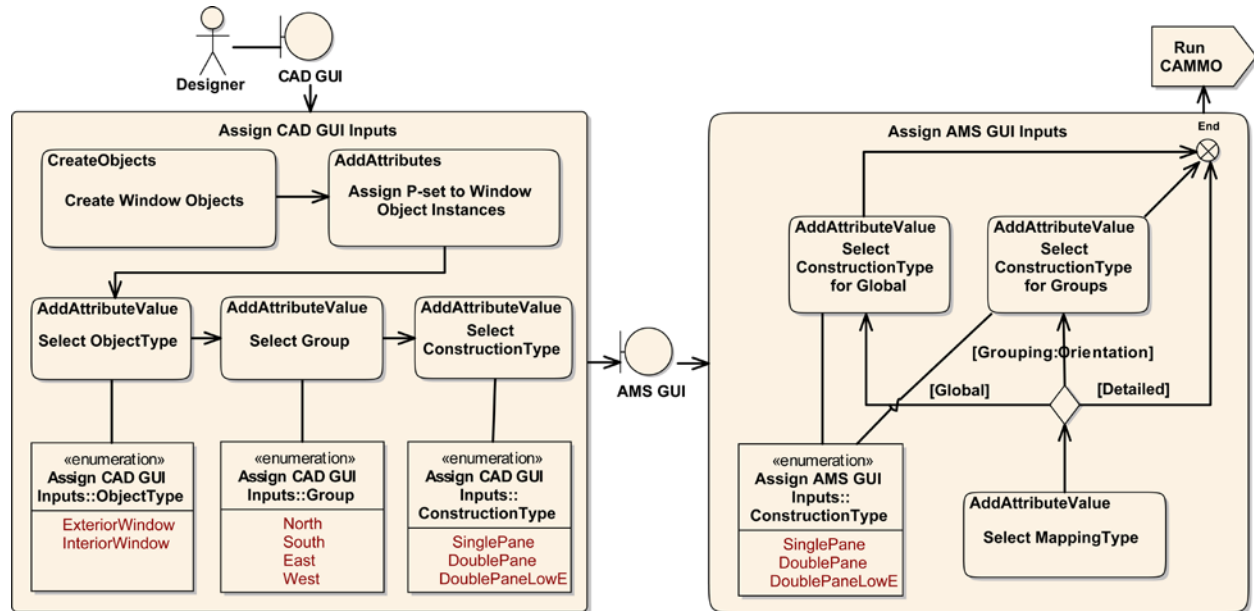
Analysis application wrappers are CAE application specific; each CAE application must have its own unique wrapper. Each analysis application wrapper accesses the AMS database, extracts the information it needs for its simulation inputs, and executes any required data transformations before generating an input file and running the simulation. The authors' implementation has enabled the wrappers to use the same geometry model and naming convention for design variables, minimizing the processing requirements prior to simulation and ensuring consistency in the analysis models.

## **2.7. Dynamic Attribution Structure**

As discussed in Section 1.4, past research has suggested the need to group different attributes for assignment to a given geometric object. However, a method to group instances of a given object type for assignment of a given object type attribute value, or dynamic attribution, is of greater importance for MDO environments. This functionality is required not just for the proper configuration of attributes modified during optimization, but also for the proper configuration of attributes that will remain static during the course of an optimization. CAMMO supports dynamic attribution with a DAS. A DAS enables designers to configure a desired problem formulation by specifying ranges of object type attribute values for various groupings of object types. The authors implemented and the designer manipulates the DAS using the CAD GUI and AMS, as was shown in Figure 2.

The authors' DAS employs the concept of *Mapping Types* to allow various mapping strategies between object type instances and object type attribute values. A Mapping Type is a mechanism to isolate a group of object type instances and assign it an object type attribute value independent of the remaining object type instances. There are three DAS Mapping Types: *Global*, *Grouping*, and *Detailed*. Using these Mapping Types, object type attribute values are assigned to either all instances of a given object type

(Global), subsets of instances of a given object type (Grouping), or a single instance of a given object type (Detailed), matching the requirements of the three attribution strategies in Figure 1. The designer can specify the unique nature of the Grouping Mapping Type, for example Grouping:Orientation, Grouping:SpaceType, Grouping:FloorLevel, etc.

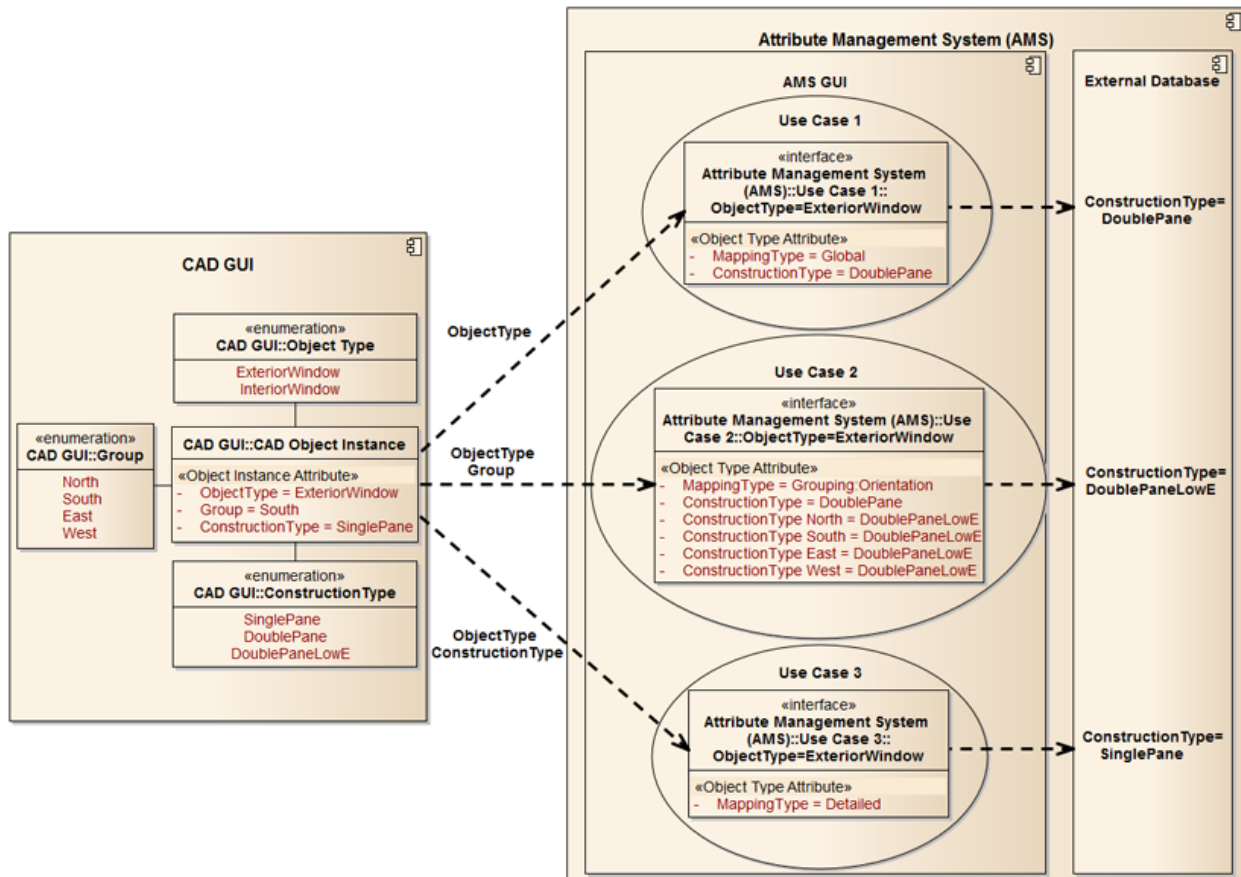


**Figure 4:** This UML diagram shows CAMMO’s user steps for window attribution. After creating window objects in the CAD model, the designer assigns custom p-sets to each window object instance in the CAD GUI and selects attribute values for ObjectType, Group, and ConstructionType. Next, the designer selects a MappingType for the object type ExteriorWindow in the AMS GUI. If the MappingType is Global, the designer selects a value for ConstructionType for all ExteriorWindow object types. If the MappingType selected is Grouping:Orientation, the designer assigns a value for ConstructionType to each of the group values (North, South, East, and West). Finally, if the designer selects the MappingType Detailed, then no further steps need to be taken. Once the AMS GUI inputs are complete, the designer can run CAMMO.

Figure 4 shows CAMMO’s user steps in the CAD and AMS GUIs. In the CAD GUI, the designer assigns a custom p-set with pre-defined non-geometric attributes to each object instance based on object family. Each p-set contains the attributes object type, group, and a range of other analysis-specific attributes for each object family that are defined by the designer. The first attribute value assigned by the user is for the attribute object type. The value selected for this attribute determines which of the remaining attributes are required, and the GUI deactivates non-applicable attributes. The designer then selects a value for the group attribute, followed by values for the remaining analysis attributes. The designer may select multiple objects at once and attributes assigned for efficiency. Initial object type attribute values are assigned in the CAD GUI prior to accessing the AMS GUI if attribute values were assigned for every object instance within the AMS, then they would all be exposed individually as design variables in the Optimization GUI. This scenario would create unmanageably large variable trees in the Optimization GUI, and only allow the designer to optimize one object instance at a time. The designer must carefully manage design variable exposure in the Optimization GUI to ensure usability.

In the AMS GUI, the designer then assigns a Mapping Type to each active object type. In this scenario, all the object type attributes inherit the same Mapping Type. However, if desired, it is also possible to assign a Mapping Type directly to an object type attribute. In this case, the Mapping Type for the child object type attribute overrides any Mapping Type assigned to the parent object type. Additionally, the designer may assign more than one Mapping Type to a single object type in the AMS GUI. For example,

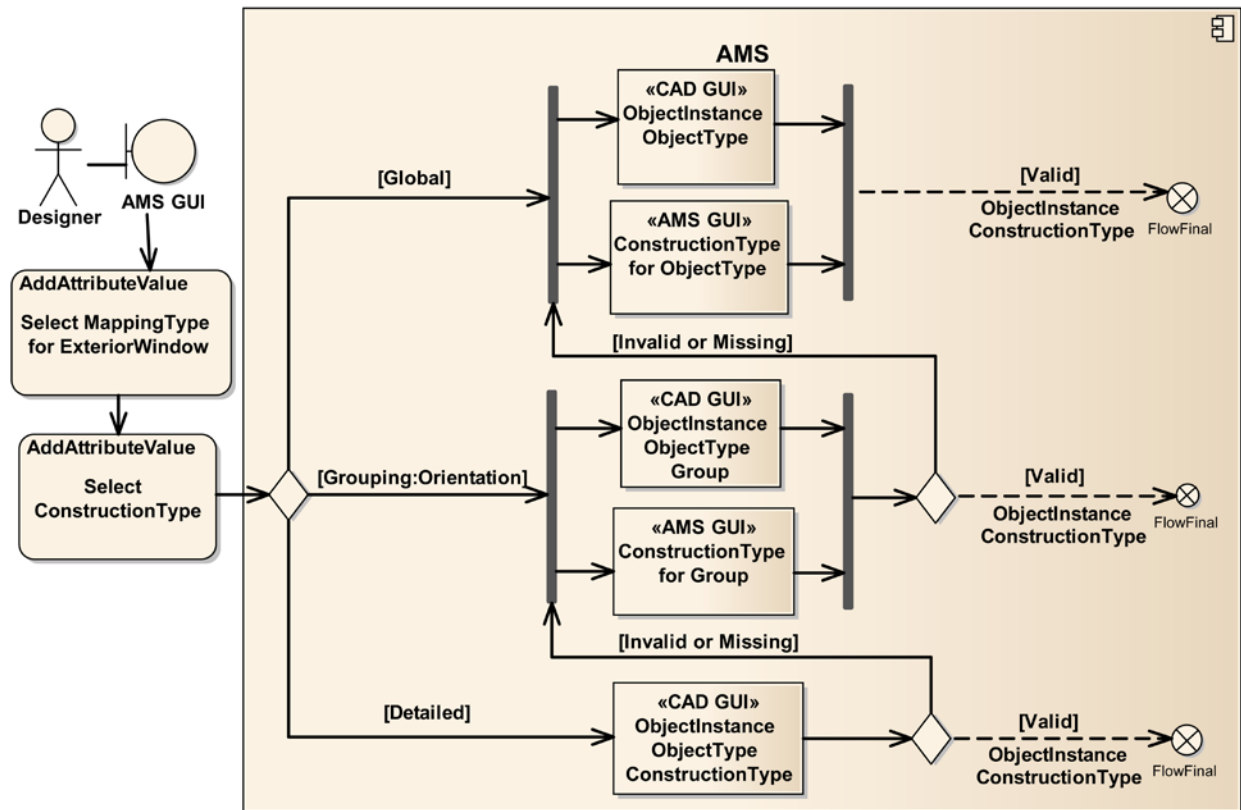
if the designer wants to isolate a group of object types located on the third floor of the south orientation, the designer could assign two object groups to that object in the CAD GUI, then assign the two Mapping Types Grouping:Orientation and Grouping:FloorLevel in the AMS GUI. After each of these user steps is completed and the designer executes the automated process, the resultant attribute values are stored in the external database with their corresponding CAD geometry. Figure 5 shows the software data flows once the user steps are completed.



**Figure 5:** This UML diagram shows the data flows of CAMMO’s DAS given three different use cases for window attribution. In use case 1, the designer defined the MappingType as Global. In this scenario, the AMS only needs the value for the attribute object type from the CAD model for each object instance. If the object type is ExteriorWindow, the value for the object type attribute ConstructionType assigned to that instance is DoublePane, and the result is stored in the database. In use case 2, the MappingType is Grouping:Orientation. In this case, both the attribute values for object type and group are used from the CAD model to map the object instance to its corresponding ConstructionType (DoublePaneLowE). In use case 3, the MappingType is Detailed. In this case, ConstructionType is extracted directly from the CAD model for all object instances assigned an object type of ExteriorWindow.

In the case that a Mapping Type does not exist for a desired group, the designer may either create a new one via the AMS and update the CAD model, or utilize a secondary mechanism of the DAS to optimize on that group of object type instances without the need to formally create a new Mapping Type. This mechanism is a system of defaulting Mapping Types. The designer can assign a “trigger value” to object attributes in the CAD GUI, such as “Null”, to engage the defaulting Mapping Types. For example, if the designer selects Grouping:Orientation as the Mapping Type for the object type ExteriorWindow in the AMS GUI, and wants to optimize for the object type attribute Construction Type on all the third floor

south-facing exterior windows, the “Null” trigger may be assigned in the CAD GUI to all the instances of south-facing windows on the floors other than the third floor in the field where the Construction Type would normally be assigned. When the optimizer iterates through various values of Construction Type on the south orientation, it will do so for all instances of south-facing ExteriorWindows except those with the trigger value, which will default to the Global Mapping Type and receive the same static value for Construction Type the designer assigned. Figure 6 shows a UML diagram of this system. This is the method used when the designer wants to optimize on a single object type instance or provide stability and fault tolerance for missing or incorrect information.



**Figure 6:** When the designer selects a high-fidelity MappingType such as Detailed, a null, missing, or invalid value for ConstructionType from the CAD model will engage an automatic shift up to the next highest-fidelity MappingType. Likewise, if a null, missing, or invalid value for either the group or ConstructionType attributes exists when using the Grouping MappingType, the system will assign the object instance the ConstructionType for the Global MappingType, which is always valid. CAMMO’s method of defaulting Mapping Types enables complete flexibility in how to attribute groups of object types, while at the same time providing a robust methodology to address missing or invalid data.

### 3. VALIDATION CASE STUDIES

The following sections describe the research validation metrics and the technical implementations, problem formulations, and results for two industry case studies. One industry case study is a large office building retrofit project (including some new construction) located in Washington, DC. The second case study is a medium border station new construction project located in Columbus, New Mexico. Both projects have aggressive sustainability goals, particularly in terms of passive thermal performance. The results for these diverse case studies provide strong evidence for CAMMO’s scalability, generality, and power.



### 3.1. Validation Metrics

Metrics to evaluate the quality of design space exploration have generally proven elusive [66, 67]. Researchers have proposed such metrics as quantity, variety, quality, and novelty to represent how well a design method explores a design space [68, 69]. However, these metrics do not specifically address the issue of flexibility in problem formulation for an automated design method. How well a problem formulation is configured directly impacts the quality of the design space exploration. Therefore, the authors propose a new metric called Problem Formulation Effectiveness (PFE) to measure flexibility in problem formulation for CAMMO, and therefore the quality of the resulting design space exploration in terms of its accuracy and cost-effectiveness. The authors define this new metric as follows:

- *Problem Formulation Effectiveness (PFE)*: PFE is defined as the ratio of the size of the alternative space evaluated for the implemented problem formulation to the size of the alternative space that would need to be evaluated for the appropriate problem formulation for a given optimization problem (OP). An OP is a set of one or more design variables and their range of values for simulation as specified by the design team. An “appropriate problem formulation” is one where the software is able to be configured to exactly match the attribution configuration desired by the design team for the OP. PFE is a measure of how well the implemented problem formulation matches the appropriate problem formulation for both individual design variables in an OP, as well as the OP as a whole. This metric provides a measure for the accuracy and cost-effectiveness of different MDO problem formulation methods. A PFE greater than 1 means that the size of the alternative space that must be evaluated is larger than is required for the posed design challenge. This scenario results in longer analysis times than are necessary, resulting in both poor accuracy and cost-effectiveness due to time requirements yet still maintaining the ability to assess the design challenge. A PFE less than 1 means that the size of the alternative space that must be evaluated is smaller than is required for the posed design challenge, signifying that the implemented problem formulation is unable to adequately assess the design challenge, thereby reducing the accuracy of the design space exploration.

While PFE serves as a *direct* metric for the impact of the problem formulation on design space exploration, the overall results of the design space exploration serve as an *indirect* metric for the impact of a CAD-centric attribution methodology, as well as the MDO design method as a whole. For example, a CAD-centric attribution methodology that allows for the appropriate problem formulation (and the resulting design space exploration accuracy and cost-effectiveness) may allow the design team to implement MDO whereas the lack of such a methodology may not. Therefore, any increase in the number of design alternatives evaluated, decrease in time per alternative evaluated, or improvement in the final design’s thermal or cost performance (among others) could reasonably be attributed to the availability of that methodology. The metric to evaluate the overall results of the design space exploration selected for this research is Product Performance (PP). The authors define this metric as follows:

- *Product Performance (PP)*: The product performance is the predicted energy and daylighting performance of the building design. For energy performance, PP is measured by annual energy costs (\$/yr) and calculated by the EnergyPlus Wrapper. For daylighting performance, PP is measured by the average Daylight Saturation (DS) [70] for perimeter spaces and calculated by the Radiance Wrapper. Daylight Saturation is a measure of the annual quantity of daylight taking into account both the duration of daylight relative to occupied hours and the amount of daylight relative to the required levels for the daylit spaces. The value is a percentage with a range of 0 -100% with 100% meaning that daylight provided 100% of the lighting requirement for 100% of the time. Daylight Saturation accounts for times of partial daylight saturation where daylight contributes to the ambient lighting but does not fully meet the requirement. The minimum DS for both case studies is 0.7. The validation



will compare the PP resulting from the MDO trade study with the PP resulting from the design team's final set of measure options.

### 3.2. Industry Case Study #1: GSA Office Building

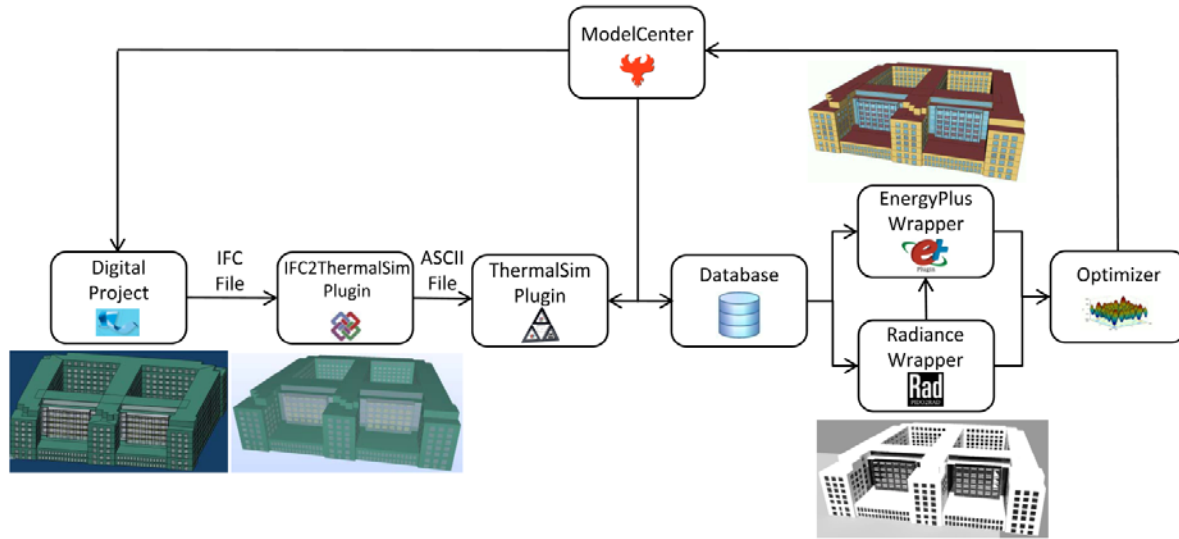
The first industry case study is a large General Services Administration (GSA) [71] office building in Washington D.C.: a 7-floor, 500,000 ft<sup>2</sup> building undergoing a major renovation to improve its energy and daylighting performance. GSA is the United States (US) federal agency that designs, constructs, and operates federal buildings for the US government, including office buildings, border stations, and courthouses. The renovation includes a new 105,000 ft<sup>2</sup> south-facing glass atrium. The project is targeting LEED Silver certification. Some of the passive thermal design strategies that will be implemented are (1) replacing the historic punched windows; (2) adding internal shading devices; (3) installing low-e vertical atrium glass filled with argon; (5) installing photovoltaic glass on the horizontal skylights; (6) adding horizontal external shading devices to the atrium wall; (7) installing daylight controls; and (8) increase wall and roof insulation. Figure 7 shows the proposed retrofit.



**Figure 7:** The GSA office building in Washington D.C is seeking a LEED Silver Certification for an upcoming major renovation, including a new 105,000 ft<sup>2</sup> south-facing atrium. (Image courtesy of GSA)

#### 3.2.1. Technical Implementation

The authors constructed a Digital Project model of the planned renovation using project drawings and a Revit model provided by the design team, and successfully exported, pre-processed, and attributed the IFC file in the ThermalSim Plugin and converted to analysis models by the EnergyPlus and Radiance Wrappers (Figure 8). The parametric model took approximately 2 weeks to build, while the conversion process to analytical models for EnergyPlus and Radiance took approximately 1 week to implement. With the technical implementation complete, the actual processing time from Digital Project to both EnergyPlus and Radiance takes under 3 minutes. Both Wrappers are able to support the above passive thermal design strategies. The building was simulated using static variable-air-volume (VAV) and fan coil heating, ventilation, and air-conditioning systems (HVAC).



**Figure 8:** The authors modeled the GSA office building industry case study in Digital Project and successfully processed it through the IFC2ThermalSim Plugin, ThermalSim Plugin, and EnergyPlus/Radiance Wrappers.

### 3.2.2. MDO Problem Formulation Requirements

To validate the capabilities of CAMMO, the authors asked the design team for the GSA office building to provide 6 different problem formulations for analysis from their project. The first 3 OPs were to parallel an analysis the design team actually conducted in some form. The final 3 OPs are problems the design team would have liked to analyze given additional time and resources. Table 2 shows the design team’s responses. When Mapping Type=Grouping and # of Groups=1, it signifies that the # of Options is only applied to a single subset of the given object type, which still requires the Grouping functionality.

OP	Measure Description	Mapping Type	Grouping Type	# of Groups	# of Options	Size of Design Space (Meas./Total)
1	1 Replace Historic Punched Windows	Grouping	PC	2	3	486
	2 Add Networked Daylight Controls	Global	-	-	2	
	3 Increase Wall Insulation	Global	-	-	3	
	4 Increase Roof Insulation	Grouping	PV/none	2	3	
2	1 Improve Atrium Vert. Glass	Grouping	Dir/ PC	1	5	225
	2 Improve Atrium Hor. Glass	Grouping	Dir	1	5	
	3 Add Green Screen to Atrium	Grouping	Dir/ PC	1	3	
	4 Improve Atrium Sunshades (Coating)	Global	-	-	3	
3	1 Improve Atrium Vert. Glass	Grouping	Dir/ PC	1	5	75
	2 Improve Atrium Hor. Glass	Grouping	Dir	1	5	
	3 Improve Atrium Sunshades (Depth)	Global	-	-	3	
4	1 Replace Historic Punched Windows	Grouping	PC	2	5	2,025
	2 Add Internal Shading Devices	Grouping	Dir/ PC	4	3	
5	1 Replace Historic Punched Windows	Grouping	PC	2	3	73,728
	2 Add Overhangs	Grouping	Dir/Fl#	6	2	
	3 Add Internal Shading Devices	Grouping	Dir/Fl#	6	2	
	4 Add Networked Daylight Controls	Global	-	-	2	
6	1 Improve Atrium Vert. Glass	Grouping	Dir/ PC	1	5	102,400
	2 Improve Atrium Hor. Glass	Grouping	Dir	1	5	
	3 Improve Atrium Sunshades (Depth)	Grouping	Fl#	6	4	

**Table 2:** This table shows the GSA office building problem formulations specified by the design team. The authors calculated the size of the design space for each measure is by taking the # of Options raised to the power of the # of Groups and the size of the design space for each OP by multiplying the measure design spaces together. Key: Dir=Direction, PC=Perimeter/Core, PV=Photovoltaic, Fl#=Floor#. The smallest design space is for OP #3 with 75 options; the largest design space is for OP # 6 with 102,400 options.

### 3.2.3. Case Study Results

To determine the PFE for each measure and OP, the authors first calculated the size of three baseline design spaces, each assuming a fixed baseline Mapping Type (Global, Grouping, and Detailed). The size of the Global baseline design space equals the # of Options from Table 2 except for the following scenario. When Mapping Type=Grouping and # of Groups=1 for a measure in Table 2, the corresponding baseline design space size in Table 3 for Mapping Type= Global is assigned a value of 1, indicating that a Global Mapping Type is unable to evaluate the measure. This approach minimizes the size of the baseline design space and corresponding PFE. The size of the Grouping baseline design space equals the # of Options raised to the power of the # of Groups. The size of the Detailed baseline design space equals the # of Options raised to the power of the # of Objects in the authors' CAD model. Next, the authors calculated the PFE for each baseline Mapping Type, as well as CAMMO, by dividing the baseline design space by the actual design space from Table 2. Table 3 shows the results.

OP	Size of Baseline Design Space			PFE				
	Global	Grouping	Detailed	Global	Grouping	Detailed	CAMMO	
1	1	3	6,561	4.5E+945	0.3	729	5.0E+944	1.0
	2	2	256	2.0E+56	1.0	128	9.8E+55	1.0
	3	3	81	6.2E+233	1.0	27	2.1E+233	1.0
	4	1	9	3.5E+09	0.3	1.0	3.9E+08	1.0
	Total	18	1.2E+09	1.9E+1245	0.037	2.6E+06	3.9E+1242	1.0
2	1	1	5	2.7E+110	0.2	1.0	5.5E+109	1.0
	2	1	5	2.5E+01	0.2	1.0	5.0E+00	1.0
	3	1	3	2.4E+75	0.33	1.0	8.1E+74	1.0
	4	3	729	1.8E+57	1.0	243	6.0E+56	1.0
	Total	3	5.5E+04	3.0E+244	0.013	243	1.3E+242	1.0
3	1	1	5	2.7E+110	0.2	1.0	5.5E+109	1.0
	2	3	729	1.8E+57	1.0	243	6.0E+56	1.0
	3	1	5	2.5E+00	0.2	1.0	5.0E+00	1.0
	Total	3	1.8E+04	1.2E+169	0.04	243	1.6E+167	1.0
4	1	5	3.9E+05	4.0E+1385	0.2	15,625	1.6E+1384	1.0
	2	3	81	4.5E+945	0.037	1.0	5.6E+943	1.0
	Total	15	3.2E+07	1.8E+2329	0.01	15,625	8.9E+2325	1.0
5	1	3	6,561	4.5E+945	0.3	729	5.0E+944	1.0
	2	1	64	5.5E+567	0.02	1.0	8.6E+565	1.0
	3	1	64	5.5E+567	0.02	1.0	8.6E+565	1.0
	4	2	256	2.0E+56	1.0	128	9.8E+55	1.0
	Total	6	6.9E+09	2.7E+2133	0.0001	93,312	3.7E+2128	1.0
6	1	1	5	2.7E+110	0.2	1.0	5.5E+109	1.0
	2	1	5	2.5E+01	0.2	1.0	5.0E+00	1.0
	3	1	4,096	1.8E+72	0.0002	1.0	4.3E+68	1.0
	Total	1	1.0E+05	1.2E+184	0.00001	1.0	1.2E+179	1.0

**Table 3:** This table shows the size of the baseline design spaces for the GSA office building and the resulting PFE for each baseline Mapping Type and CAMMO. Baseline assumptions for Grouping are Fl# for overhangs and Dir. for all other object types.

The first five OPs contain a mixture of desired Mapping Types on the measure level. This diversity results in overall PFEs much less than 1.0 for the Global baseline Mapping Type and much larger than 1.0 for the Grouping and Detailed baseline Mapping Types. Consequently, had an automated process only allowed for the Global Mapping Type, the design team would have been able to evaluate only a small fraction of their desired design space, thereby risking the selection of sub-optimal designs due to inaccurate design space exploration. Alternatively, had an automated process only allowed for the Grouping or Detailed Mapping Types, the size of the required design spaces would have been unnecessarily large (in most cases prohibitively large), resulting in inaccuracy and low cost-effectiveness for the design space exploration.

For example, in OP #1, the total PFE of 0.037 for the Global Mapping Type reflects the limited design space exploration available to the design team (18 of 486 alternatives). However, had only a Grouping or Detailed Mapping Type been implemented without the ability to evaluate individual measures globally, then the resulting size of the design spaces that would need to be evaluated would be 2.6E+06 and 3.9E+1242 times larger than was required, respectively. The time required to evaluate design spaces of these magnitudes would drastically reduce the cost-effectiveness of the MDO process. The likely result would be for the design team to only evaluate a small fraction of the required design space, resulting in low accuracy as well as low cost-effectiveness. As CAMMO enables the designer to modify the Mapping Type to match each measure problem formulation configuration, the PFE for each measure and OP is 1.0.

To evaluate the potential impact of using a process such as CAMMO for the design of the GSA office building, the authors and the project team formulated the optimization problem shown in Table 4 to be executed using the authors’ technical implementation. An additional measure of adding blast drapes [72] to the perimeter punched windows was included upon request of the project team. The authors chose a design of experiments (DoE) to evaluate the OP design space rather than an optimization to allow for the simulation and visualization of performance trends over the entire spectrum of the design space, and to prevent any potential deficiencies in the selected optimization algorithm from skewing the results due to the large number of discrete variables. Table 5 shows the baseline design spaces and resulting PFE for each measure.

#	Measure Description	Mapping Type	Grouping Type	# of Groups	# of Options	Size of Design Space (Meas./Total)
1	Replace Historic Punched Windows	Grouping	PC	2	3	23,328
2	Increase Wall Insulation	Global	-	-	3	
3	Increase Roof Insulation	Global	-	-	3	
4	Improve Atrium Vert. Glass	Grouping	Dir/ PC	1	4	
5	Improve Atrium Hor. Glass	Grouping	Dir	1	4	
6	Improve Atrium Sunshades (Depth)	Grouping	Fl#	2	3	
7	Add Blast Drapes to Punched Windows	Grouping	PC	1	2	

**Table 4:** This table lists the measures evaluated for the GSA office building DoE with their corresponding configurations. Appendix A shows the details of the measure options (enumerations).

#	Size of Baseline Design Space			PFE			
	Global	Grouping	Detailed	Global	Grouping	Detailed	CAMMO
1	3	6,561	4.5E+945	0.3	729	5.0E+944	1.0
2	3	81	6.2E+233	1.0	27	2.1E+233	1.0
3	3	3	3.5E+09	1.0	1.0	1.2E+09	1.0
4	1	4	1.3E+95	0.25	1.0	3.3E+94	1.0
5	1	4	1.6E+01	0.25	1.0	4.0E+00	1.0
6	1	729	1.8E+57	0.11	81	2.0E+56	1.0
7	1	16	4.3E+419	0.5	8.0	2.2E+419	1.0
Total	27	3.0E+11	1.6E+1762	0.0012	1.3E+07	2.2E+1758	1.0

**Table 5:** This table shows the size of the baseline design spaces for the GSA office building DoE and the resulting PFE for each baseline Mapping Type and CAMMO.

To demonstrate the power of CAMMO’s fully-automated process to be implemented within the time constraints typical of building conceptual design, the design space was simulated using distributed and parallel computing resources [73-75]. To enable the use of a wide-range of computing platforms with minimal technical challenges, the authors developed a methodology to use Model Center and CAMMO to pre-process a compressed file for each design iteration in the DoE that includes the appropriate EnergyPlus and Radiance Wrapper input files, with separate compressed files for the executables for each analysis application. Using this approach, the designer may schedule the DoE over any distributed and parallel computing resource, cloud-based or otherwise, by simply running a parametric sweep of every compressed file in the DoE. The method copies and extracts the EnergyPlus and Radiance executables locally to each compute node as needed to minimize the size of the DoE compressed files uploaded and downloaded from each computing server. The results returned from the computing resource are then parsed, post-processed, and visualized within Model Center. This method eliminates the need to install and manage the Model Center and Analysis Server applications over the distributed network, which in some cases, causes significant technical and run time difficulties.

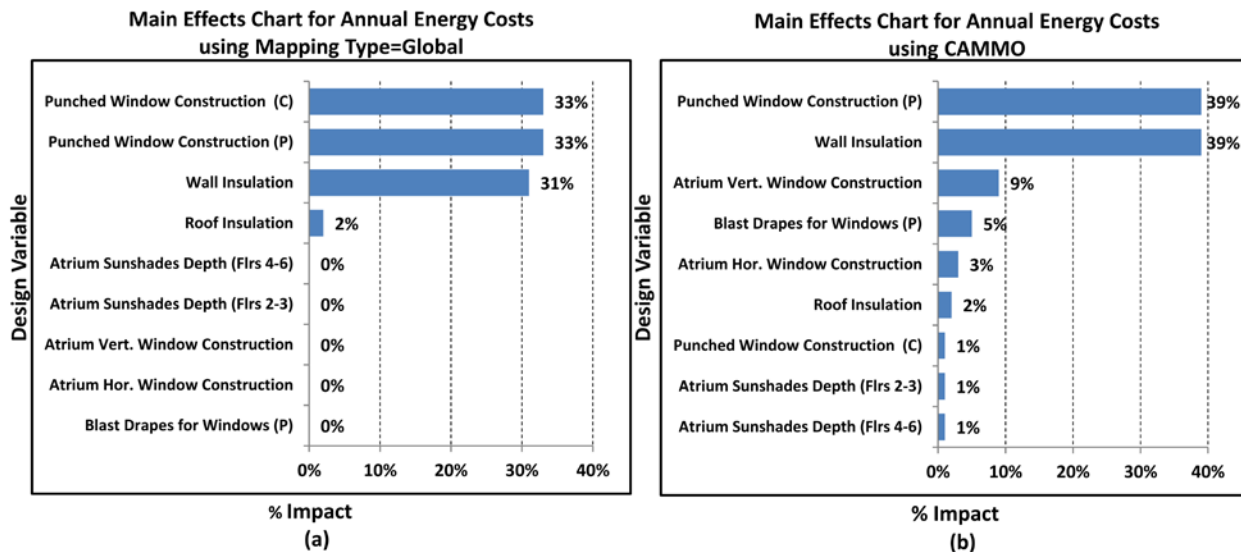
The simulations for the DoE were run in parallel using the Windows Azure cloud computing platform [76] and a 7000-node Linux-based computing cluster at Stanford’s High Performance Computing Center (HPCC) [77]. At any given time, 4000 nodes were running in parallel on Azure and 1000 nodes on the HPCC cluster, resulting in 5000 total simulations run in parallel. The generation of the 23,328 compressed input file directories took approximately 8 hours, while the post-processing of the results for all the runs took 0.3 hours. The results in Table 6 show that the Azure and HPCC computing resources simulated the entire design space of 23,328 alternatives in under 20 hours. The average run time and total run time are averages between the two clusters. EnergyPlus and Radiance ran significantly faster in the Linux-based environment with a decrease in run time of approximately 30%. The best performing design from the DoE that met the minimum daylighting performance constraint resulted in a decrease in annual operating costs for energy of 16.5% relative to the design team’s final design. As expected, the daylighting performance decreased by 11.4%, for all the options evaluated to decrease conductive, convective, and radiative thermal loads decreased the amount of visible light penetrating the perimeter spaces by design.

Avg. Run Time (hrs)	# Parallel Cores	Total Run Time (hrs)	PP (\$/yr) (millions)			PP (DS)		
			Final Design	CAMMO	% Change	Final Design	CAMMO	% Change
4.2	5000	19.6	\$1.97	\$1.64	-16.5%	0.79	0.70	-11.4%

**Table 6:** The table shows the results of the GSA office building DoE, including computational requirements and PP (annual energy cost and daylight saturation), for both the project team’s final design and the best performing design

using CAMMO. The simulation times are averages between the Windows Azure cloud computing platform and Stanford's HPCC cluster.

To demonstrate the impact of PFE on this example, the resulting variable influence on annual energy costs had the design team been limited to a Mapping Type of Global is shown in Figure 9a. Figure 9b shows the actual variable influence on annual energy costs given the appropriate problem formulation in Table 4 using CAMMO. A comparison of the two figures reveals that had the design team been limited to a Global Mapping Type, the analysis results would have provided incorrect information as to the relative impact of the design variables on annual energy costs, resulting in poor accuracy of the design space exploration (only 27 of 23,328 alternatives evaluated). Had the design team been limited to a Grouping Mapping Type, they could have obtained the same results as Figure 9b (once the results were filtered), but at the expense of a total run time of  $2.5 \times 10^8$  hours (28,767 years) using the same computing resources shown in Table 6.



**Figure 9:** This figure shows the relative impacts of the design variables on annual energy costs for the GSA office building (a) for a Mapping Type of Global and (b) using CAMMO. Without the availability of CAMMO, the resulting problem formulation would result in inaccurate prioritization of variable importance on annual energy costs, as well as different relative magnitudes of variable impacts.

This increase in total run time is obviously prohibitively large and drastically lowers (if not eliminates) the cost-effectiveness of the design space exploration. If the design team had been limited in execution time for this DoE to one month of total run time on 5,000 cores, only 0.0003% of the design space would have been explored, resulting in not only lower cost-effectiveness, but decreased accuracy as well for there would be no means to determine if the best performing designs were simulated within this timeframe.

### 3.3. Industry Case Study #2: GSA Border Station

The second industry case study is a US border station in Columbus, New Mexico. The GSA is designing the 24,000 ft<sup>2</sup> new construction project to be the first NZE building in GSA's portfolio of buildings and is seeking LEED Platinum Certification. The design incorporates a wide range of passive thermal design strategies, many of them targeted at taking advantage of the large diurnal temperature swings at the site. Passive design strategies include (1) solar chimneys to induce stack ventilation through a tempered underground thermal storage reservoir; (2) solar air preheaters for the solar chimneys; (3) nighttime purge; (4) thermal mass inside the insulation on roof, floor, and wall surfaces; (5) a sawtooth roof with

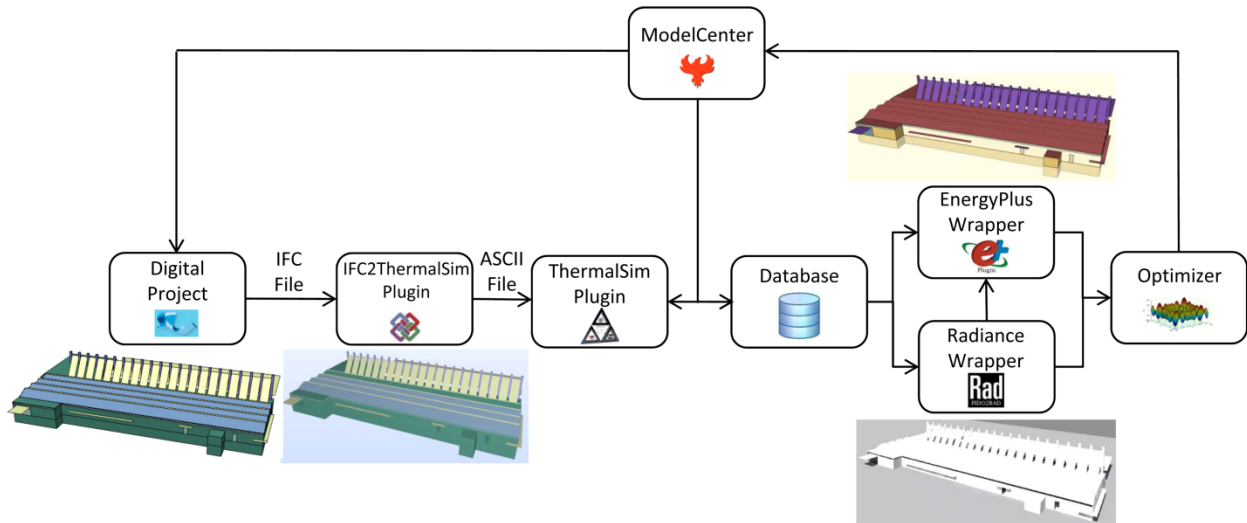
diffusing shades, baffles, and photovoltaic panels; and (6) various daylighting strategies. Figure 10 shows a profile of the current border station design.



**Figure 10:** A new US border station located in New Mexico will be the first federal NZE building. The figure shows the proposed solar chimneys that provide natural stack ventilation during the day through the basement thermal reservoir and allow for night flushing to pre-cool the building. (Image courtesy of Richter Architects)

### 3.3.1. Technical Implementation

The authors constructed a Digital Project model using project drawings and a Revit model provided by the design team, and successfully exported, pre-processed, and attributed the IFC file in the ThermalSim Plugin and converted it to analysis models using the EnergyPlus and Radiance Wrappers (Figure 11). The parametric model took approximately 1 week to build, while the conversion process to analytical models for EnergyPlus and Radiance took approximately 1 week to implement. With the technical implementation complete, the actual processing time from Digital Project to both EnergyPlus and Radiance takes under 1 minute. Both Wrappers are able to support the above passive thermal design strategies except strategies (1) and (2) due to limitations in EnergyPlus. The building was simulated using a static variable-air-volume (VAV) HVAC system coupled with ground-source heat pumps.



**Figure 11:** The authors successfully modeled the GSA border station in Digital Project and processed it through the IFC2ThermalSim Plugin, ThermalSim Plugin, and EnergyPlus/Radiance Wrappers.

### 3.3.2. MDO Problem Formulation Requirements

The authors gave the design team for the border station the same optimization problem formulation guidelines for this research as the GSA office building project team in Section 3.2.2. Table 7 shows the design teams' responses.

OP	Measure Description	Mapping Type	Grouping Type	# of Groups	# of Options	Size of Design Space (Measure/Total)
1	1 Improve Ext. Wall Construction	Grouping	Dir	4	3	81
	2 Improve Floor Slab Construction	Global	-	-	2	2
	3 Improve Ext. Window Construction	Grouping	Dir	3	2	8
2	1 Modify Orientation	Global	-	-	3	3
	2 Change PV Tilt	Global	-	-	3	3
	3 Change Window Height	Grouping	Dir	3	3	27
3	1 Reduce Lighting Density	Grouping	ST	4	2	16
	2 Modify Illuminance Setpoint	Grouping	ST	4	2	16
	3 Improve Ext. Window Construction	Grouping	Dir	3	2	8
4	1 Change Overhang Depth	Grouping	Dir	3	3	27
	2 Change Window Height	Grouping	Dir	3	3	27
	3 Improve Ext. Window Construction	Grouping	Dir	5	3	243
5	1 Change PV Tilt	Global	-	-	5	5
	2 Improve Ext. Window Construction	Grouping	Dir	1	4	4
	3 Add Interior Window Shading	Grouping	Dir	1	4	4
	4 Change Skylight Overhang Depth	Grouping	Dir	1	4	4
	5 Change Skylight Window Height	Grouping	Dir	1	5	5
6	1 Improve Roof Slab Construction	Global	-	-	5	5
	2 Improve Ext. Wall Construction	Grouping	Dir	4	5	625
	3 Improve Glass Door Construction	Grouping	Dir	2	3	9

**Table 7:** This table shows the GSA border station problem formulations specified by the design team. The authors calculated the size of the design space for each measure is by taking the # of Options raised to the power of the # of Group and the size of the design space for each OP by multiplying the measure design spaces together. Key: Dir=Direction, ST=Space Type. The smallest design space is for OP #2 with 243 options; the largest design space is for OP # 4 with 177,147 options.

### 3.3.3. Case Study Results

The PFE for each measure and OP was determined using the same method described in Section 3.2.3 for the GSA office building case study. Table 8 shows the results.

OP	Size of Baseline Design Space			PFE				
	Global	Grouping	Detailed	Global	Grouping	Detailed	CAMMO	
1	1	3	81	1.2E+19	0.04	1.0	1.5E+17	1.0
	2	2	4	2.1E+06	1.0	2.0	1.0E+06	1.0
	3	2	8	5.2E+05	0.25	1.0	6.6E+04	1.0
	Total	12	2,592	1.3E+31	0.01	2.0	1.0E+28	1.0
2	1	3	3	3.0E+00	1.0	1.0	1.0E+00	1.0
	2	3	9	2.0E+04	1.0	3.0	6.6E+03	1.0
	3	3	27	1.2E+09	0.11	1.0	4.3E+07	1.0
	Total	27	729	6.9E+13	0.11	3.0	2.8E+11	1.0
3	1	2	16	2.1E+06	0.13	1.0	1.3E+05	1.0
	2	2	16	5.1E+02	0.13	1.0	3.2E+01	1.0
	3	2	8	5.2E+05	0.25	1.0	6.6E+04	1.0
	Total	8	2,048	5.6E+14	0.004	1.0	2.7E+11	1.0
4	1	3	27	5.9E+04	0.11	1.0	2.2E+03	1.0
	2	3	27	1.2E+09	0.11	1.0	4.3E+07	1.0
	3	3	243	1.2E+09	0.01	1.0	4.8E+06	1.0



	Total	27	1.8E+05	8.0E+22	0.0002	1.0	4.5E+17	1.0
5	1	5	25	2.0E+06	1.0	5.0	3.9E+05	1.0
	2	1	4	2.6E+05	0.25	1.0	6.6E+04	1.0
	3	1	4	2.6E+05	0.25	1.0	6.6E+04	1.0
	4	1	4	1.0E+03	0.25	1.0	2.6E+02	1.0
	5	1	5	2.0E+06	0.20	1.0	3.9E+05	1.0
	Total	5	8,000	2.7E+26	0.003	5.0	1.7E+23	1.0
6	1	5	25	6.1E+09	1.0	5.0	1.2E+09	1.0
	2	5	625	9.5E+13	0.01	1.0	1.5E+11	1.0
	3	3	9	1.968E+04	0.33	1.0	2.2E+03	1.0
	Total	75	1.4E+05	1.1E+28	0.003	5.0	4.1E+23	1.0

**Table 8:** This table shows the size of the baseline design spaces for the GSA border station and the resulting PFE for each baseline Mapping Type and CAMMO. Baseline assumptions for Grouping are Dir and ST.

As was the result in the first case study, four of the six OPs contain a mixture of desired Mapping Types on the measure level. OP#3 and OP#4 only use the Grouping Mapping Type. As in first case study, CAMMO’s PFE for each measure and OP is 1.0. The implications of a Global or Detailed baseline Mapping Type for this project are similar to those in the previous example in terms of accuracy and cost-effectiveness of the design space exploration. However, the implications of a Grouping baseline Mapping Type for this project are quite different. The PFE in this case for the mixed Mapping Type OPs ranges from 2.0-5.0, quite small compared to the PFEs in the first example for mixed Mapping Type OPs (243-2.6E+06). This scenario results in minimal penalty on the design space exploration in terms of cost-effectiveness, and possibly no penalty in terms of accuracy since the DoE for these values of PFE would most likely be able to be carried out in its entirety. The contrast in the impact of a Grouping baseline Mapping Type between the two case studies demonstrates the high degree of variability between projects for problem formulation requirements and the consequences of a rigid automated process.

To evaluate the potential impact of using a process such as CAMMO for the design of the GSA border station, the authors and the project team formulated the optimization problem shown in Table 9 to be executed using the authors’ technical implementation. As with the first case study, the authors chose a design of experiments (DoE) to evaluate the OP design space rather than an optimization to allow for the simulation and visualization of performance trends over the entire spectrum of the design space, and to prevent any potential deficiencies in the selected optimization algorithm from skewing the results due to the large number of discrete variables. Table 10 shows the baseline design spaces and resulting PFE for each measure.

#	Measure Description	Mapping Type	Grouping Type	# of Groups	# of Options	Size of Design Space (Measure/Total)
1	Change Window Height	Grouping	Dir	1	3	69,120
2	Change Window Width	Grouping	Dir	1	2	
3	Change Skylight Overhang Depth	Grouping	Dir	1	2	
4	Change Skylight Window Height	Grouping	Dir	1	3	
5	Improve Ext. Window Construction	Grouping	Dir	2	4	
6	Improve Glass Door Construction	Global	-	-	3	
7	Improve Ext. Wall Construction	Global	-	-	4	
8	Improve Roof Slab Construction	Global	-	-	5	
9	Add Interior Window Shading	Grouping	Dir	1	2	

**Table 9:** This table lists the measures evaluated for the GSA border station DoE with their corresponding configurations. Appendix B shows the details of the measure options (enumerations).

#	Size of Baseline Design Space			PFE			
	Global	Grouping	Detailed	Global	Grouping	Detailed	CAMMO
1	1	3	3.0E+00	0.33	1.0	1.0E+00	1.0
2	1	2	2.0E+00	0.5	1.0	1.0E+00	1.0
3	1	2	3.2E+01	0.5	1.0	1.6E+01	1.0
4	1	3	2.4E+02	0.33	1.0	8.1E+01	1.0
5	4	16	2.7E+11	0.25	1.0	1.7E+10	1.0
6	3	27	2.0E+04	1.0	3.0	6.6E+03	1.0
7	4	256	1.2E+24	1.0	64.0	3.0E+23	1.0
8	5	25	6.1E+09	1.0	5.0	1.2E+09	1.0
9	1	2	3.2E_01	0.5	1.0	1.6E+01	1.0
Total	240	6.6E+07	6.0E+55	0.003	960	8.6E+50	1.0

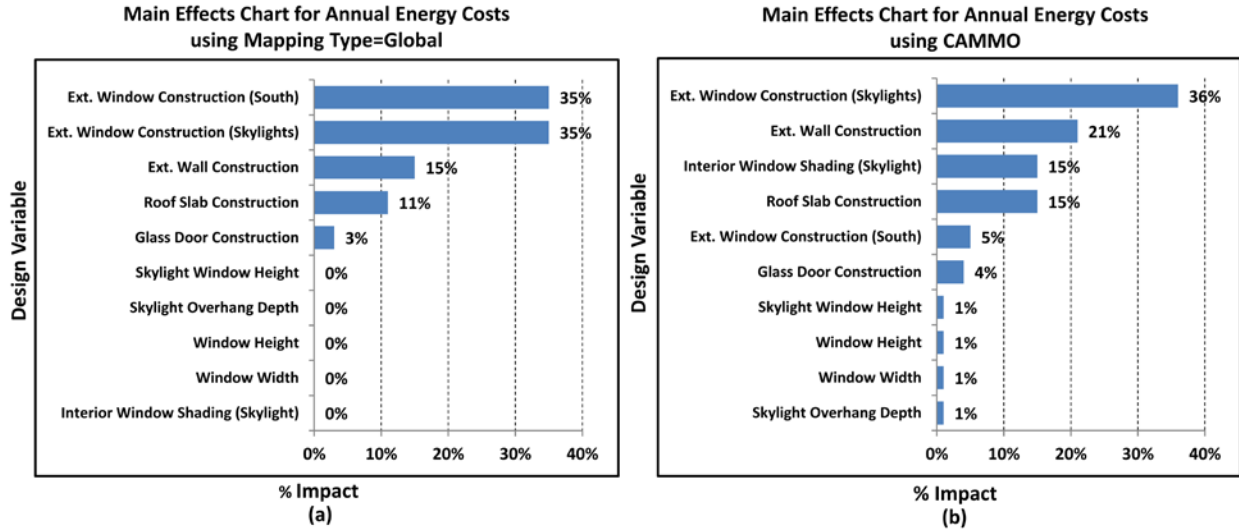
**Table 10:** This table shows the size of the baseline design spaces for the GSA border station DoE and the resulting PFE for each baseline Mapping Type and CAMMO.

The authors simulated the entire design space of 69,120 alternatives using the distributed/parallel computing methodology and Stanford HPCC Linux cluster described in the previous case study. The generation of the 69,120 compressed input file directories took approximately 6 hours, while the post-processing of the results for all the runs took 0.5 hours. The total run time for the DoE was approximately 35 hours. Table 11 shows the remaining results. Annual operating costs for energy decreased by 96% relative to the design team’s final design, and daylighting performance increased by 17%. The low absolute values for the annual operating costs is attributed to the simulations accounting for significant photovoltaic energy generation while not accounting for the high exterior lighting load, which is the dominate load at the border station. Had exterior lighting been included in the simulations, the absolute change in PP for annual operating costs would remain the same, though the percent change would be significantly lower.

Avg. Run Time (hrs)	# Parallel Cores	Total Run Time (hrs)	PP (\$/yr)			PP (DS)		
			Final Design	CAMMO	% Change	Final Design	CAMMO	% Change
0.5	1,000	34.6	\$7,693	\$340	-96%	0.88	0.96	+17%

**Table 11:** This table shows the results of the GSA border station DoE, including computational requirements and PP (annual energy cost and daylight saturation), for both the project team’s final design and the best performing design using CAMMO. The authors ran the simulations for this DoE on Stanford’s HPCC cluster.

To demonstrate the impact of PFE on this example, the resulting variable influence on annual energy costs had the design team been limited to a Mapping Type of Global is shown in Figure 12a. Figure 12b shows the actual variable influence on annual energy costs given the appropriate problem formulation in Table 9. A comparison of the two figures reveals that had the design team been limited to a Global Mapping Type, the analysis results would have provided incorrect information as to the relative impact of the design variables on annual energy costs, resulting in poor accuracy of the design space exploration (only 240 of 69,120 alternatives evaluated). Had the design team been limited to a Grouping Mapping Type, they could have obtained the same results as Figure 12b (once the results were filtered), but at the expense of a total run time of 33,000 hours (3.8 years) using the same computing resources shown in Table 11.



**Figure 12:** This figure shows the relative impacts of the design variables on annual energy costs for the GSA border station (a) for a Mapping Type of Global and (b) using CAMMO. Without the availability of CAMMO, the resulting problem formulation would result in inaccurate prioritization of variable importance on annual energy costs, as well as different relative magnitudes of variable impacts.

This increase in total run time is obviously prohibitively large and drastically lowers (if not eliminates) the cost-effectiveness of the design space exploration. Even with an increase in the number of parallel cores to 10,000, the trade study would take 4.6 months to complete. If the design team had been limited in execution time for this DoE to one month of total run time on 10,000 cores, only 22% of the design space would have been explored, resulting in not only lower cost-effectiveness, but decreased accuracy as well for there would be no means to determine if the best performing designs were simulated within this timeframe.

#### 4. DISCUSSION AND CONCLUSIONS

This paper presents a methodology for CAD-centric attribution for MDO called CAMMO. The theoretical contribution of this research is a method for CAD-centric attribution using a CAD GUI, a geometry file export using a neutral file format, a pre-processor, external attribute management system, analysis application wrappers, and a dynamic attribution structure that enables flexible MDO problem formulation. The authors tested CAMMO on two diverse and complex industry case studies: a retrofit of a large office building and a new medium-sized border station. The office and border station projects differ by project type (retrofit vs. new construction), building type (office vs. border station), size (large vs. medium), climate (mixed-humid vs. hot-dry), sustainability goals (LEED Silver vs. NZE), and passive thermal design strategies (construction type vs. orientation and geometric dimensions), respectively. The wide range of PFE and consequences of the baseline Mapping Types in terms of accuracy and cost-effectiveness for their respective OPs provides evidence that problem formulation configuration requirements can vary considerably between different projects, design firms, and measures. CAMMO's successful technical implementation and application to these case studies with such diverse physical and problem formulation requirements is evidence for the method's scalability and generality.

The authors demonstrate the power of CAMMO and its impact on practice with the results of the two industry case studies, which illustrate the improved accuracy, cost-effectiveness, and product performance realizable with CAMMO's ability to flexibly formulate a CAD-centric MDO problem. This research also shows how the use of distributed and parallel computing can dramatically reduce the computational time

requirements to evaluate CAMMO-enabled MDO processes. Further validation with additional case studies in the AEC industry is necessary to identify potential limitations of the methodology beyond the two case studies conducted for this research.

The authors' intuition is that CAMMO can be applied to a wide range of design and analysis domains, however a contribution is only claimed in the domain of PTD in AEC using energy and daylighting simulation tools as the method was not validated in other domains or with other analysis tools for building performance analysis. Further research is needed to test the methodology's effectiveness for non-thermal and non-AEC applications.

Dynamic attribution is only one component of flexible problem formulation for CAD-centric MDO. Future research will explore flexible problem formulation for CAD-centric MDO in terms of methods to automate the analysis configuration for an optimization. Specifically, the authors plan to develop an intelligent, performance-based mechanism for product decomposition to automate the identification of building spaces in a product model that have unique daylighting profiles and warrant simulation for daylighting performance. The research will include strategies to scale the results of the simulated spaces to the rest of the CAD model, or product recomposition. Such methods would result in more accurate, rapid, and consistent problem formulation capabilities for daylighting simulation, with or without MDO.

## **5. ACKNOWLEDGEMENTS**

This research is funded by the Center for Integrated Facility Engineering (CIFE), Precourt Energy Efficiency Center (PEEC), and the Civil and Environmental Engineering Department at Stanford University. Microsoft and Stanford IT Services also contributed to the HPC implementation. The authors would also like to thank the following contractors for their software development support: Chi Ng at Gehry Technologies for his support in improving the IFC export of Digital Project, Matthias Weise at AEC3 and Hannu Lahtela at Granlund for their contributions to the IFC2ThermalSim Plugin, Zack Rogers at Daylighting Innovations for his contributions to the Radiance Wrapper, Grant Soremekun and Mike Haisma at Phoenix Integration for their contributions to the EnergyPlus Wrapper, and Steve Roach at Microsoft for his HPC/Windows Azure support. The authors would also like to thank the GSA for their contribution of the industry case studies for this research.

## **6. REFERENCES**

- [1] I. Watson, S. Perera, Case-based design: A review and analysis of building design applications, *Journal of Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, 11 (1997) 59-87.
- [2] D.S. Lazzara, CAD-Based Multidexterity Analysis and Multidisciplinary Optimization in Aircraft Conceptual Design, PhD Thesis Proposal, Department of Aeronautics & Astronautics, Aerospace Computational Design Lab, Massachusetts Institute of Technology, Cambridge, MA, 2008, pp. 37.
- [3] M.P. Gallaher, A.C. O'Conner, J.L. Dettbarn, Jr., L.T. Gilday, Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry, National Institute of Standards and Technology (NIST), Gaithersburg, MD, 2004, pp. 1-194.
- [4] K. Shea, R. Aish, M. Gourtovaia, Towards integrated performance-driven generative design tools, *Automation in Construction*, 14 (2005) 253-264.
- [5] O. Akin, Variants of design cognition, in: C. Eastman, W. Newstetter, M. McCracken (Eds.), *Design Knowing and Learning: Cognition in Design Education*, Elsevier, New York, NY, 2001, pp. 105-124.
- [6] E. Ipek, S.A. McKee, R. Caruana, B.R.d. Supinski, M. Schulz, Efficiently exploring architectural design spaces via predictive modeling, *SIGOPS Oper. Syst. Rev.*, 40 (2006) 195-206.

- [7] F. Flager, B. Welle, P. Bansal, G. Soremekun, J. Haymaker, Multidisciplinary Process Integration and Optimization of a Classroom Building, *Journal of Information Technology in Construction (ITCon)*, 14 (2009) 595-612.
- [8] T. Kenworthy, CAD-Centric Dynamic Workflow Creation, Masters Thesis, Department of Mechanical Engineering, Brigham Young University, Provo, UT, 2009, pp. 1-69.
- [9] S. Subrahmanyam, M.J. Pratt, W. DeVries, Feature attributes and their role in product modeling, Proceedings of the third ACM symposium on Solid modeling and applications, ACM, Salt Lake City, Utah, United States, 1995, pp. 115-124.
- [10] V. Bradshaw, *The Building Environment: Active and Passive Control Systems*, 3rd ed., John Wiley & Sons, Inc., Hoboken, NJ, 2006.
- [11] H.M. Lechner, *Heating, Cooling and Lighting*, 3rd ed., John Wiley & Sons, Inc., Hoboken, NJ, 2009.
- [12] A. Kwok, W. Grondzik, *The Green Studio Handbook: Environmental Strategies for Schematic Design*, Elsevier, Amsterdam, 2007.
- [13] S. Mendler, W. Odell, M.A. Lazarus, *The HOK Guidebook to Sustainable Design*, 2nd ed., John Wiley & Sons, Inc., Hoboken, NJ, 2006.
- [14] B. Givoni, *Passive and Low Energy Cooling of Buildings*, John Wiley & Sons, Inc., New York, NY, 1994.
- [15] B. Givoni, *Climate Considerations in Building and Urban Design*, John Wiley & Sons, Inc., New York, NY, 1998.
- [16] W. Wang, R. Zmeureanu, H. Rivard, Applying multi-objective genetic algorithms in green building design optimization, *Building and Environment*, 40 (2005) 1512-1525.
- [17] J.P.T.J. Berends, M.J.L. van Tooren, MDO Design Support by Integrated Engineering Services within a Multi-Agent Task Environment, 26th International Congress of the Aeronautical Sciences, Anchorage, Alaska, 2008, pp. 1-18.
- [18] P. Geyer, Component-oriented decomposition for multidisciplinary design optimization in building design, *Advanced Engineering Informatics*, 23 (2009) 12-31.
- [19] A. Isaacs, K. Sudhakar, P.M. Mujumdar, Design and Development of MDO Framework, International Conference on Modeling, Simulation, Optimization for Design of Multi-disciplinary Engineering Systems, Goa, India, 2003, pp. 1-5.
- [20] S.A. O., T.J. C., Framework Requirements for MDO Application Development, Report #: AIAA-98-4740, NASA Langley Research Center, Hampton, VA, 1998.
- [21] M. Mourshed, D. Kelliher, M. Keane, ArDOT: A tool to optimize environmental design of buildings, Eighth International IBPSA Conference, Eindhoven, Netherlands, 2003, pp. 919-926.
- [22] J. Haymaker, J. Kunz, B. Suter, M. Fischer, Perspectives: Composable, Reusable Reasoning Modules To Construct an Engineering View from Other Engineering Views., *Advanced Engineering Informatics*, 18 (2004) 49-67.
- [23] I. Kroo, Distributed Multidisciplinary Design and Collaborative Optimization, S. University (Ed.), VKI lecture series on Optimization Methods & Tools for Multicriteria/Multidisciplinary Design, 2004, pp. 1-22.
- [24] F. Flager, Interactive Collaborative Optimization - A Multidisciplinary Optimization Method Applied to the Design of Steel Building and Civil Structures, CIFE Working Paper, Stanford University, 2009.
- [25] E. Eason, J. Wright, Implementation of Non-Hierarchic Decomposition for Multidisciplinary System Optimization, AIAA/NASA/USAF/OAI 4th Symposium on Multidisciplinary Analysis and Optimization, Cleveland, OH, 1992, pp. 1008-1016.
- [26] J. Sobieszczanski-Sobieski, J. Agte, R.J. Sandusky, Bi-level integrated system synthesis (BLISS), AIAA/USAF/NASA/ISSMO 7th Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, 1998.
- [27] DOE, DOE2 Homepage, EnerLogic and James J. Hirsch and Associates, 2009. Accessed in January, from: <http://www.doe2.com/>.

- [28] L. Caldas, Generation of energy-efficient architecture solutions applying GENE\_ARCH: An evolution-based generative design system, *Advanced Engineering Informatics*, 22 (2008) 59-70.
- [29] C. Christensen, R. Anderson, S. Horowitz, A. Courtney, J. Spencer, BEopt(TM) Software for Building Energy Optimization: Features and Capabilities, Report #: NREL/TP-550-39929, National Renewable Energy Laboratory, 2006, pp. 1-21.
- [30] P.G. Ellis, B.T. Griffith, N. Long, P. Torcellini, D. Crawley, Automated Multivariate Optimization Tool for Energy Analysis, NREL/CP-550-40353, National Renewable Energy Laboratory, 2006.
- [31] M. Wetter, BuildOpt--a new building energy simulation program that is built on smooth models, *Building and Environment*, 40 (2005) 1085-1092.
- [32] M. Wetter, Design optimization with GenOpt, *Building Energy Simulation News*, 21 (2000) 19-28.
- [33] K. Papamichael, J. LaPorta, H. Chauvet, Building Design Advisor: Automated Integration of Multiple simulation Tools, *Automation in Construction*, 6 (1997) 341-352.
- [34] A. Mahdavi, A comprehensive computational environment for performance based reasoning in building design and evaluation, *Automation in Construction*, 8 (1999) 427-435.
- [35] K.P. Lam, N.H. Wong, L.J. Shen, A. Mahdavi, E. Leong, W. Solihin, K.S. Au, Z. Kang, Mapping of industry building product model for detailed thermal simulation and analysis, *Advances in Engineering Software*, 37 (2006) 133-145.
- [36] M. Fischer, Formalizing Construction Knowledge for Concurrent Performance-Based Design, in: I. Smith (Ed.), *Intelligent Computing in Engineering and Architecture*, Springer Berlin / Heidelberg, 2006, pp. 186-205.
- [37] C. Eastman, P. Teicholz, R. Sacks, K. Liston, *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors*, John Wiley & Sons, Inc., Hoboken, NJ, 2008.
- [38] C.A. Crawford, R. Haimes, Synthesizing an MDO Architecture in CAD, D.o.A.a.A.M.I.o. Technology (Ed.), 42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2004, pp. 1-11.
- [39] J.C. Townsend, J.A. Samareh, R.P. Weston, W.E. Zorumski, Integration of a CAD System into an MDO Framework, N.L.R. Center (Ed.), Hampton, VA, 1998, pp. 1-9.
- [40] U. Gabbert, P. Wehner, The product data model as a pool for CAD-FEA data, *Engineering with Computers*, 14 (1998) 115-122.
- [41] M.L. King, M.J. Fisher, C.G. Jensen, A CAD-centric Approach to CFD Analysis With Discrete Features, *Computer-Aided Design & Applications*, 3 (2006) 279-288.
- [42] W.F. Bronsvort, F.W. Jansen, Feature modelling and conversion--Key concepts to concurrent engineering, *Computers in Industry*, 21 (1993) 61-86.
- [43] S. Myung, S. Han, Knowledge-based parametric design of mechanical products based on configuration design method, *Expert Systems with Applications*, 21 (2001) 99-107.
- [44] M.S. Shephard, P.M. Finnigan, Integration of geometric modeling and advanced finite element preprocessing, *Finite Elements in Analysis and Design*, 4 (1988) 147-162.
- [45] S. Arabshahi, D.C. Barton, N.K. Shaw, Steps towards CAD-FEA integration, *Engineering with Computers*, 9 (1993) 17-26.
- [46] M.L. King, A CAD-Centric Approach to CFD Analysis with Discrete Features, Masters Thesis, Department of Mechanical Engineering, Brigham Young University, Provo, UT, 2004, pp. 1-133.
- [47] J.K. Shelley, Incorporating Computational Fluid Dynamics into the Preliminary Design Cycle, Masters Thesis, Department of Mechanical Engineering, Brigham Young University, Provo, UT, 2005, pp. 1-94.
- [48] K. London, V. Singh, N. Gu, C. Taylor, L. Brankovic, Towards the Development of a Project Decision Support Framework for Adoption of an Integrated Building Information Model using a Model Server, in: *Handbook of Research on Building Information Modeling and Construction Informatics: Concepts and Technologies*, IGI Global, 2010.
- [49] T. Baker, Attribution Standardization for Integrated Concurrent Engineering, Masters Thesis, Department of Mechanical Engineering, Brigham Young University, Provo, UT, 2005, pp. 1-123.

- [50] R. O'Bara, M. Beall, M. Shepard, Attribution Management for Engineering Analysis, *Engineering with Computers*, 18 (2002) 339-351.
- [51] M.S. Shephard, The specification of physical attribute information for engineering analysis, *Engineering with Computers*, 4 (1988) 145-155.
- [52] T. Kenworthy, C.G. Jensen, CAD-Centric Dynamic Workflow Generation, *Computer-Aided Design & Applications*, 6 (2009) 673-683.
- [53] V. Bazjanac, A. Kiviniemi, Reduction, Simplification, Translation and Interpretation in the Exchange of Model Data., D. Rebolj (Ed.), 24th Conference on Bringing ITC Knowledge to Work, Faculty of Civil Engineering, University of Maribor, Moribar, Slovenia, 2007, pp. 163-168.
- [54] C. Eastman, J.-m. Lee, Y.-s. Jeong, J.-k. Lee, Automatic rule-based checking of building designs, *Automation in Construction*, 18 (2009) 1011-1033.
- [55] I. Phoenix Integration, Design Exploration and Optimization solutions: Tools for Exploring, Analyzing, and Optimizing Engineering Designs, Technical White Paper, Blacksburg, VA, 2004, pp. 1-20.
- [56] Gehry, Digital Project Documentation, Gehry Technologies, Los Angeles, CA, 2011. Accessed in January, from: [www.gtwiki.org](http://www.gtwiki.org).
- [57] DOE, EnergyPlus Energy Simulation Software, US Department of Energy, Washington, D.C., 2011. Accessed in January from: <http://apps1.eere.energy.gov/buildings/energyplus/>.
- [58] DOE, Radiance Homepage, US Department of Energy, Washington, D.C., 2011. Accessed in January from: <http://radsite.lbl.gov/>.
- [59] B. Welle, J. Haymaker, Z. Rogers, ThermalOpt: A methodology for automated BIM-based multidisciplinary thermal simulation for use in optimization environments, *Building Simulation*, 4 (2011) 293-313.
- [60] M.S. Shephard, M.W. Beall, R.M. O'Bara, B.E. Webster, Toward simulation-based design, *Finite Elements in Analysis and Design*, 40 (2004) 1575-1598.
- [61] C.M. Eastman, *Building Product Models: Computer Environments Supporting Design and Construction*, CRC Press, Inc., Boca Raton, FL, 1999.
- [62] J. Kim, H. Soonhung, Manipulating Geometry in a STEP DB from Commercial CAD Systems, *Concurrent Engineering: Research and Applications*, 12 (2004) 49-57.
- [63] buildingSMART, Industry Foundation Classes (IFC) Homepage, M.S. Group (Ed.), 2011. Accessed in January from: <http://buildingSMART-tech.org>.
- [64] H.-F. Wang, Y.-L. Zhang, CAD/CAM integrated system in collaborative development environment, *Robotics and Computer-Integrated Manufacturing*, 18 (2002) 135-145.
- [65] M.L. Maher, J.H. Rutherford, A model for synchronous collaborative design using CAD and database management, *Research in Engineering Design*, 9 (1997) 85-98.
- [66] H.A. Bashir, V. Thomson, Metrics for design projects: a review, *Design Studies*, 20 (1999) 263-277.
- [67] L.C. Briand, S. Morasca, V.R. Basili, Defining and validating measures for object-based high-level design, *IEEE Transactions on Software Engineering*, 25 (1999) 722-743.
- [68] J.J. Shah, S.M. Smith, N. Vargas-Hernandez, Metrics for measuring ideation effectiveness, *Design Studies*, 24 (2003) 111-134.
- [69] H. McManus, M. Richards, A. Ross, D. Hastings, A Framework for Incorporating "ilities" in Tradespace Studies, AIAA Space 2007, Long Beach, CA, 2007.
- [70] C.F. Reinhart, J. Mardaljevic, Z. Rogers, Dynamic daylight performance metrics for sustainable building design, *Leukos*, 3 (2006) 1-25.
- [71] P.B.S. General Services Administration (GSA), GSA Homepage, 2010. Accessed in, from: [www.gsa.gov](http://www.gsa.gov).
- [72] Safetydrape Homepage. Accessed in March 2012, from: <http://safetydrape.com/>.
- [73] Y. Khalidi, Building a Cloud Computing Platform for New Possibilities, *Computer*, 44 (2011) 29-34.
- [74] F. Iorio, J.L. Snowdon, Leveraging Cloud Computing and High Performance Computing Advances for Next-generation Architecture, *Urban Design and Construction Projects, 2011 Symposium on Simulation for Architecture and Urban Design*, ACM, Orlando, FL, 2011, pp. 69-76.

- [75] F. Flager, A. Adya, J. Haymaker, AEC Multidisciplinary Design Optimization: Impact of High Performance Computing, CIFE Technical Report #TR186, Center for Integrated Facility Engineering (CIFE), Stanford University, 2009.
- [76] D. Chappell, Windows HPC Server and Windows Azure, David Chappell & Associates and Microsoft, 2010, pp. 1-17.
- [77] Stanford, Stanford High Performance Computing Center. Accessed in February 2012, from: <http://hpcc.stanford.edu>.



**Appendix A**  
**GSA Office Building DoE Measure Enumerations**

#	Measure Description	Measure Enumerations
1	Replace Historic Punched Windows	1800F_EWin_Street, 1800F_EWin_Courtyard, DP_LowE_Arg_1, DP_LowE_Arg_2
2	Increase Wall Insulation	1800F_EWall, 1800F_EWall_6inRB, 1800F_EWall_8inRB
3	Increase Roof Insulation	1800F_RoofSlab_Floor7, 1800F_RoofSlab_Floor7_R50, 1800F_RoofSlab_Floor7_R60
4	Improve Atrium Vertical Glass	1800F_EWin_VA_Street, DP_DowE_Arg_1, DP_DowE_Arg_2, DP_DowE_Kry_2
5	Improve Atrium Horizontal Glass	1800F_EWin_HA, DP_DowE_Arg_1, DP_DowE_Arg_2, DP_DowE_Kry_2
6	Improve Atrium Sunshades (Depth)	0.8m, 1.2m, 1.5m
7	Add Blast Drapes to Punched Windows	None, Safetydrape

**Appendix B**  
**GSA Border Station DoE Measure Enumerations**

#	Measure Description	Measure Enumerations
1	Change Window Height	0.3m, 0.45m, 0.6m
2	Change Window Width	21m, 30m
3	Change Skylight Overhang Depth	0.5m, 0.8m
4	Change Skylight Window Height	0.5m, 1.0m, 1.5m
5	Improve Ext. Window Construction	T&G_Sungate_500_Optigray23, DP_LowE_Arg_1, DP_LowE_Arg_2, DP_LowE_Kry_2
6	Improve Glass Door Construction	T&G_Sungate_500_Optigray23, DP_LowE_Arg_1, DP_LowE_Kry_2
7	Improve Ext. Wall Construction	Prop_Wall_Const, Prop_Wall_Const_R30, Prop_Wall_Const_R35, Prop_Wall_Const_R40
8	Improve Roof Slab Construction	Roof_Construction, Roof_Construction_R30, Roof_Construction_R40, Roof_Construction_R50, Roof_Construction_R60
9	Add Interior Window Shading	WindowShadingOn, WindowShadingOff