Proceedings of Renewable and Advanced Energy Systems for the
21st Century
April 11-15, 1999, Lahaina, Maui, Hawaii

LBNL-42734

EnergyPlus:
A New-Generation Building Energy Simulation Program

Drury B. Crawley
U.S. Department of Energy
Washington, DC

Linda K. Lawrie
U.S. Army Construction Engineering Research Laboratories
Champaign, Illinois

Curtis O. Pedersen, Richard J. Liesen, Daniel E. Fisher, Richard K. Strand,
and Russell D. Taylor
University of Illinois at Urbana-Champaign
Urbana, Illinois

Frederick C. Winkelmann, W. F. Buhl,
Y. Joe Huang, and A. Ender Erdem
Lawrence Berkeley National Laboratory
Berkeley, California

April 1999

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technology, State and Community Programs, Office of Building Systems of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

EnergyPlus: A New-Generation Building Energy Simulation Program

Drury B. Crawley
U.S. Department of Energy
Washington, DC

Linda K. Lawrie
U.S. Army Construction Engineering Research Laboratories
Champaign, Illinois

Curtis O. Pedersen, Richard J. Liesen, Daniel E. Fisher, Richard K. Strand, and Russell D. Taylor
University of Illinois at Urbana-Champaign
Urbana, Illinois

Frederick C. Winkelmann, W. F. Buhl, Y. Joe Huang, and A. Ender Erdem
Lawrence Berkeley National Laboratory
Berkeley, California

Abstract
Many of the popular building energy simulation programs around the world are reaching maturity –some use simulation methods (and even code) that originated in the 1960s. For more than two decades, the U.S. government supported development of two hourly building energy simulation programs, BLAST and DOE–2. Designed in the days of mainframe computers, expanding their capabilities further has become difficult, time-consuming, and expensive. At the same time, the 30 years have seen significant advances in analysis and computational methods and power—providing an opportunity for significant improvement in these tools.

In early 1996, a Federal agency began developing a new building energy simulation tool, EnergyPlus, building on development experience with DOE–2 and BLAST. EnergyPlus includes a number of innovative simulation features—such as variable time steps, built-in template and external modular systems that are integrated with a heat balance-based zone simulation—and input and output data structures tailored to facilitate third party module and interface development. Other planned simulation capabilities include multi-zone airflow, and electric power and solar thermal and photovoltaic simulation. Beta testing of EnergyPlus begins in mid 1999.

Introduction
For the past 20 years, the U.S. government supported development of DOE–2 and BLAST. The BLAST program [BLA 92] was sponsored by the U.S. Department of Defense (DOD) and had its origins in the NBSLD program developed at the U.S. National Bureau of Standards (now NIST) in the early 1970s. The DOE–2 program [WIN 93] was sponsored by the U.S. Department of Energy (DOE) and had its origins in the Post Office program written in the late 1960s for the U.S. Post Office. The main difference between the programs is the load calculation method—DOE–2 uses a room weighting factor approach while BLAST uses a heat balance approach. Both programs are widely used throughout the world.

Each program comprises hundreds of subroutines working together to simulate heat and mass energy flows throughout a building. In some cases, subroutines in DOE–2 were more accurate. In other cases, subroutines in BLAST were more accurate. In both programs, however, simulation methodologies (or loops) are often difficult to trace due to decades of development (and multiple authors). Often, this results in "spaghetti code" with data and subroutines for a particular simulation capability spread throughout the program. To modify either program, a developer must have many years of experience working within the code, knowledge of code unrelated to their task (because of the spaghetti), and (for sponsors) an extraordinary investment of time and money.

Why the U. S. government supported two separate (and comparable capability) programs has been questioned for many years. Discussions on merging the two programs began in earnest in 1994 with a DOD-sponsored workshop in Illinois. Nothing concrete resulted from that workshop, but eventually, DOE took the initiative and began developing a new program, named EnergyPlus, in 1996. The EnergyPlus team includes U. S. Army Construction Engineering Research Laboratories (CERL), University of Illinois (UI), Lawrence Berkeley National Laboratory (LBNL), and DOE. In this paper, we present an overview of the organization and capabilities of EnergyPlus and explain the rationale and structure behind the overall program.

Why a New Program?

As mentioned earlier, DOE–2 and BLAST have become expensive to maintain, modify and enhance, because of 20+ year-old code and old Fortran structures (or general lack of structure). When DOD ended support for BLAST in 1995 due to budget constraints, we took the opportunity to combine the resources, teams, and best capabilities and features of BLAST and DOE–2. The last version of BLAST was released in spring 1998 and the last version of DOE–2 with contributions under DOE-sponsorship was completed in 1998. Initially, we thought that we could create a "best of" program—combining modules from the two programs—without starting from scratch. After initial development work, we determined that EnergyPlus would cost less to develop, be released faster, and be easier to modify and extend if we wrote all new, modular, structured code. Thus, EnergyPlus is all-new Fortran 90 code.

What is EnergyPlus?

EnergyPlus is a new building performance simulation program that combines the best capabilities and features from BLAST and DOE–2 along with new capabilities. EnergyPlus comprises completely new code written in Fortran 90. It is primarily a simulation engine—there is no interface. Input and output are simple comma-separated, ASCII text files, a much simpler input structure than either DOE–2 or BLAST. Both BLAST and DOE–2 have been successful in attracting third-party developers for user interfaces and new modules. We have invited these same developers to participate in EnergyPlus beginning during beta testing—to work on new simulation modules or user interfaces.

Modular Code

One of the main goals for the EnergyPlus development effort has been to create a well-organized, modular structure that facilitates adding features and links to other programs. In evaluating programming languages, we found we had two choices—move to C/C++ or stay with Fortran. Despite the advantages of the structure and object-orientation of C/C++, we decided to select Fortran 90 as the programming language for EnergyPlus because Fortran 90:

- is a modern, modular language with good compilers on many platforms
- allows C-like data structures and mixed language modules
- provides structure that begins to be object-based
- allows long variable names (up to 32 characters)
- provides backward compatibility during the development process

We began working on EnergyPlus by modularizing (restructuring) code from the heat balance engine in IBLAST, a research version of BLAST with integrated loads and HVAC calculation [TAY 90, 91]. Normally such restructuring would result in major rewrites involving a long development period, and very extensive testing to ensure the new code performs as intended. However, because the EnergyPlus team selected Fortran 90 (and Fortran 77 is a subset of Fortran 90), development is proceeding through a process which we call Evolutionary Reengineering (ER). This process incrementally moves the program from old unstructured legacy code to new modular code by incorporating new code with old. The existing code still works with user input data, and is extended to generate parameters needed by the new code modules. In this way the new modules can be verified without having to completely replace the entire functional capability of the old program with new code before it can be tested. As the process proceeds, the parameters supplied by old routines are replaced by new routines and data structures. This makes the transition evolutionary and permits a smooth transition with a greater capability for verification testing.

EnergyPlus Structure

In two recent workshops on next generation energy tools sponsored by DOE and DOD [CRA 97] there was strong consensus that a more flexible and robust tool with additional capabilities was needed. Recurrent themes for simulation needs throughout the workshops were design, environment, economics, and occupant comfort and safety. Designers need tools that provide answers to very specific questions during design. They want tools that provide the highest level of simulation accuracy and detail reasonably possible but that don't get in the user's way. One of the highest priorities was an integrated (simultaneous) simulation for accurate temperature and comfort prediction.

In response to these findings, we decided that integrated simulation should be the underlying concept for EnergyPlus—loads calculated (by a heat balance engine) at a user-specified time step (15-minute default) are passed to the building systems simulation module at the same time step. The building systems simulation module, with a variable time step (down to seconds), calculates heating and cooling system and plant and electrical system response. Feedback from the building systems simulation module on loads not met is reflected in the next time step of the load calculations in adjusted space temperatures if necessary.

By using an integrated solution technique in EnergyPlus, the most serious deficiency of the BLAST and DOE–2 sequential simulations can be solved—inaccurate space temperature predication due to no feedback from the HVAC module to the loads calculations. Accurate prediction of space temperatures is crucial to energy efficient system engineering—system size, plant size, occupant comfort and occupant health are dependent on space temperatures.

Integrated simulation also allows users to evaluate a number of processes that neither BLAST nor DOE–2 can simulate well. Some of the more important include:

- Realistic system controls
- Moisture adsorption and desorption in building elements
- Radiant heating and cooling systems
- Interzone air flow

Figure 1 shows the overall program structure. EnergyPlus has three basic components—a simu-lation manager, a heat balance simulation module, and a building systems simulation module. The simulation manager controls the entire simulation process. The heat balance calculations are based on IBLAST—a research version of BLAST with integrated HVAC systems and building loads simulation.

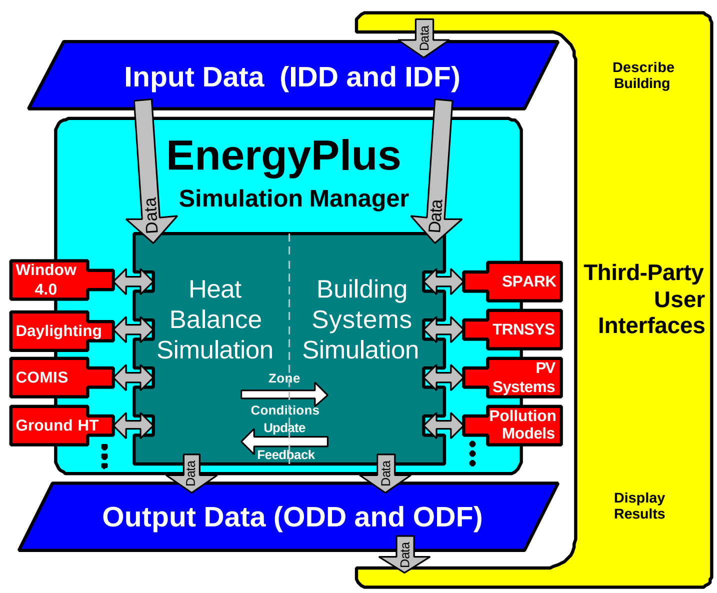

Figure 1: Overall EnergyPlus Structure

A new building systems simulation manager handles communication between the heat balance engine and various HVAC modules and loops, such as coils, boilers, chillers, pumps, fans, and other equipment/ components. (In the first release, the building systems simulation manager only has HVAC systems and equipment / components. Future releases of EnergyPlus will include elec-trical systems simulation.). The building systems simulation manager also controls inter-action and data exchange between EnergyPlus and SPARK [BUH 93] and HVACSIM+ [MET 95] simulations. Gone are the hardwired "template" systems (VAV, Constant Volume Reheat, etc.) of DOE–2 and BLAST—they are replaced by user-configured heating and cooling equipment components formerly within the template. This gives users much more flexibility in matching their simulation to the actual system configurations. The building systems simulation module also manages data communication between the HVAC modules, input data, and output data structures.

A comparison of major features and capabilities of EnergyPlus, BLAST, IBLAST, and DOE–2 are shown in Tables 1, 2, and 3. Table 1 shows general features, Table 2 shows load calculation features, and Table 3 shows HVAC features.

Table 1: Comparison of General Features and Capabilities

General Feature	DOE-2	BLAST	IBLAST	EnergyPlus
Integrated, Simultaneous Solution • Integrated loads/systems/plant • Iterative solution • Tight coupling	No	No	**Yes**	**Yes**
Multiple Time Step Approach • User-defined time step for interaction between zones and environment (15-minute default) • Variable time-step for interactions between zone air mass and HVAC system (≥ 1 minute)	No	No	**Yes**	**Yes**
Input Functions • Users can modify code without recompiling	**Yes**	No	No	**Yes**
New Reporting Mechanism • User-definable reports	No	No	No	**Yes**

Table 2: Comparison of Loads Features and Capabilities

Loads Feature	DOE-2	BLAST	IBLAST	EnergyPlus
Heat Balance Calculation • Simultaneous calculation of radiation and convection processes each time step	No	**Yes**	**Yes**	**Yes**
Interior Surface Convection • Dependent on temperature and air flow • Internal thermal mass	No **Yes**	**Yes** **Yes**	**Yes** **Yes**	**Yes** **Yes**
Moisture Absorption/Desorption • Combined heat and mass transfer in building envelopes	No	No	**Yes**	**Yes**
Thermal Comfort • Human comfort model based on activity, inside drybulb, humidity, and radiation	No	**Yes**	**Yes**	**Yes**
Anisotropic Sky Model • Sky radiance depends on sun position for better calculation of diffuse solar on tilted surfaces	**Yes**	No	No	**Yes**
Advanced Fenestration Calculations • Controllable window blinds • Layer-by-layer glazing input • Electrochromic glazing	**Yes** **Yes** **Yes**	No **Yes** **No**	No **Yes** **No**	**Yes** **Yes** **Yes**
WINDOW 4 Library • More than 200 window types—conventional, reflective, low-E, gas-fill, electrochromic • User defined using WINDOW 4	**Yes**	**Yes**	**Yes**	**Yes**
Daylighting Illumination and Controls • Interior illuminance from windows and skylights • Stepped, dimming, on/off luminaire controls • Glare simulation and control • Effects of dimming on heating and cooling	**Yes**	No	No	**Yes**

Table 3: Comparison of HVAC Features and Capabilities

HVAC Systems and Equipment Feature	DOE-2	BLAST	IBLAST	EnergyPlus
Fluid Loops • Connect primary equipment and coils • Hot water loops, chilled water and condenser loops, refrigerant loops	**Yes**	No	No	**Yes**
Air Loops • Connect fans, coils, mixing boxes, zones	No	No	No	**Yes**
User-configurable HVAC systems	No	No	No	**Yes**
Hardwired Template HVAC systems	**Yes**	**Yes**	**Yes**	No
High-Temperature Radiant Heating • Gas/electric heaters, wall radiators	No	**Yes**	No	**Yes**
Low-Temperature Radiant Heating/Cooling • Heated floor/ceiling • Cooled ceiling	No	No	**Yes**	**Yes**
Atmospheric Pollution Calculation • CO_2, SO_x, NO_x, CO, particulate matter and hydrocarbon production • On-site and at power plant • Calculate reductions in greenhouse gases	**Yes**	**Yes**	No	**Yes**
SPARK Connection	No	No	No	**Yes**
TRNSYS Connection	No	No	No	**Yes**

Simulation Management

At the outermost program level, the Simulation Manager (shown schematically in Figure 2) controls the interactions between all simulation loops from a sub-hour level up through the user selected time step and simulation period—whether day, month, season, year or several years. Actions of individual simulation modules are directed by the simulation manager, instructing simulation modules to take actions such as initialize, simulate, record keep, or report.

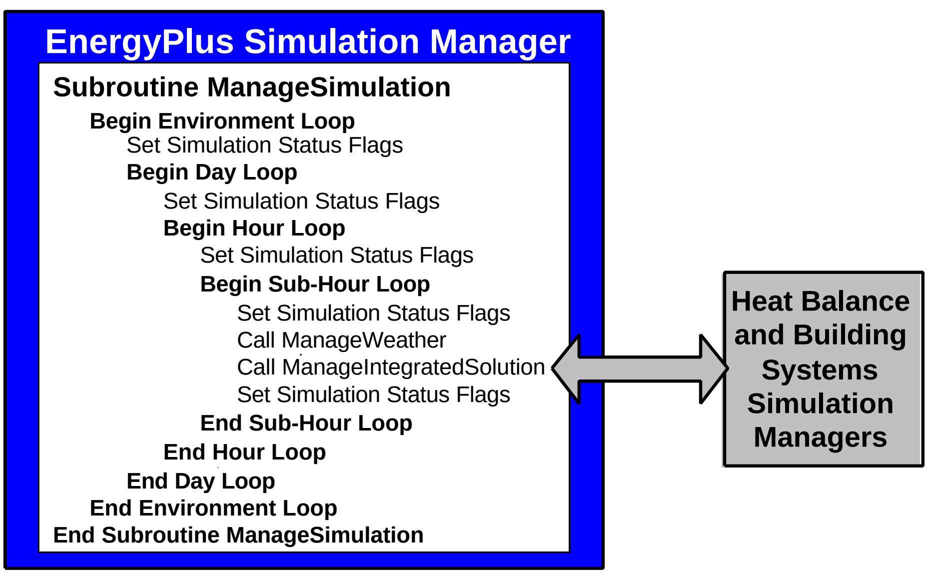

Figure 2: EnergyPlus Simulation Manager

Figure 3 shows the structure of the EnergyPlus integrated solution manager that manages the surface and air heat balance modules and acts as an interface between the heat balance and the building systems simulation manager. The surface heat balance module simulates inside and outside surface heat balance, interconnections between heat balances and boundary conditions, conduction, convection, radiation, and mass transfer (water vapor) effects. The air mass balance module deals with various mass streams such as ventilation air, exhaust air, and infiltration. It accounts for thermal mass of zone air and evaluates direct convective heat gains. Through this module that we are connecting to COMIS [FEU 90] for improved multi-zone airflow, infiltration, indoor contaminant, and ventilation calculations.

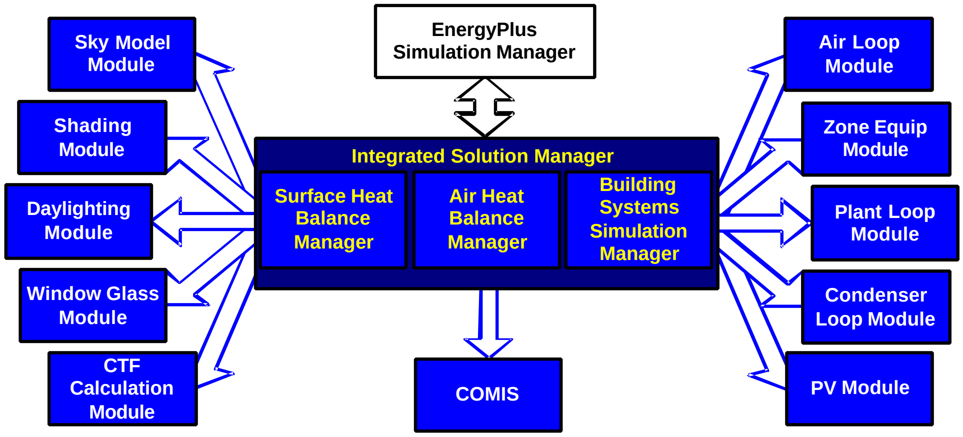

Figure 3: Integrated Simulation Manager

We created the simulation manager to specifically address the legacy issues of spaghetti code and lack of structure in DOE–2 and BLAST. The simulation manager provides several critical benefits:

- major simulation loops are contained in a single module
- modules are self-contained and more object-based
- data access is controlled
- new modules are easily added

Heat and Mass Balance

As noted earlier, the underlying building thermal zone calculation method in EnergyPlus is a heat balance model. The fundamental assumption of heat balance models is that air in each thermal zone can be modeled as well stirred with uniform temperature throughout. Although this does not reflect physical reality well, the only current alternative is Computational Fluid Dynamics (CFD)—a complex and computationally intensive simulation of fluid (in this case, air) movement. Currently, CFD is most useful in research applications. Several groups are developing models somewhere between the well-stirred model and a full CFD calculation. The modular structure of EnergyPlus allows these new models to be included in future releases once they are available. The other major assumption in heat balance models is that room surfaces (walls, windows, ceilings, and floors) have:

- uniform surface temperatures,
- uniform long and short wave irradiation,
- diffuse radiating surfaces, and
- internal heat conduction.

In addition to the basic heat balance engine from IBLAST, we are creating three new modules based on capabilities within DOE–2: daylighting illumination [WIN 85], WINDOW 4-based fenestration [ARA 94], and anisotropic sky. The daylighting module calculates hourly interior daylight illuminance, glare from windows, glare control, electric lighting controls (on/off, stepped and continuous dimming), and electric lighting reduction for the heat balance module. In the future, the daylighting module will include an improved interior inter-reflection calculation, light shelves, roof monitors, and reflection from neighboring buildings. The fenestration module includes capabilities from WINDOW 4—accurate angular dependence of transmission and absorption for both solar and visible radiation, and temperature-dependent U-value. Users can enter a layer-by-layer window description or choose windows from the library (conventional, reflective, low-e, gas fill electrochromic). Also simulated are movable window shades for sun and/or glare control. In the near future, the algorithms will be upgraded to the WINDOW 5 algorithms for coatings and framing elements. The sky model includes non-isotropic radiance and luminance distribution throughout the sky based on the empirical model by Perez as a function of sun position and cloud cover. This non-uniform radiance distribution improves calculation of diffuse solar on tilted surfaces (walls and sloped roofs).

Several other modules have been re-engineered for inclusion in EnergyPlus: solar shading from BLAST and conduction transfer function calculations from IBLAST. The major enhancements of the IBLAST (and EnergyPlus) heat balance engine over BLAST include mass transfer and radiant heating and cooling. The mass transfer capability within EnergyPlus allows fundamental, layer-by-layer solution for mass transfer through surfaces and a mass balance on zone air similar to the air heat balance. The radiant heating and cooling models are an expansion of the conduction transfer function and incorporate thermal comfort calculations. This provides a means for improved modeling and control capabilities for the new building systems simulation manager.

One last important feature of the EnergyPlus heat balance engine is that it is essentially identical in functionality to the Loads Toolkit being developed by UI under ASHRAE RP-987. UI is developing both the Loads Toolkit and the EnergyPlus heat and mass balance engine and is using the programming standard developed in the EnergyPlus project to produce the Loads Toolkit. Both projects benefit—modularization efforts started by EnergyPlus will be useful in the Loads Toolkit and new component models developed for the Loads Toolkit will enhance EnergyPlus.

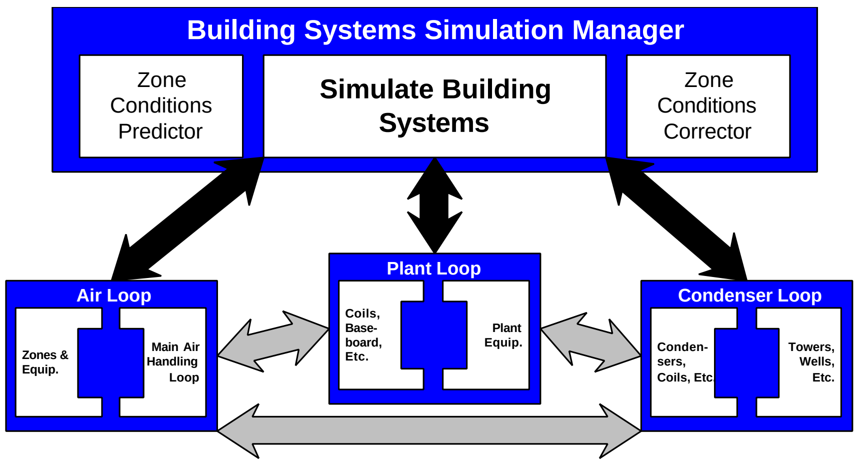

Figure 4: Building Systems Simulation Manager

Building Systems Simulation Manager

After the heat balance manager completes simulation for a time step, it calls the Building Systems Simulation Manager (see Figure 4) which controls the simulation of HVAC and electrical systems, equipment and components and updates the zone-air conditions. EnergyPlus does not use a sequential simulation method (first building loads, then air distribution system, and then central plant) as found in DOE–2 and BLAST since this imposes rigid boundaries on program structures and limits input flexibility. Instead, we designed the building systems simulation manager with several objectives in mind:

- fully integrated simulation of loads, systems, and plant
- modular
- extensible

Integrated simulation models capacity limits more realistically and tightly couples the air and water side of the system and plant. Modularity is maintained at both the component and system level. This eases adding new components and flexibly modeling system configurations and, at the system level, equipment and systems are clearly connected to zone models in the heat balance manager. To implement these concepts, we use loops throughout the building systems simulation manager—primarily HVAC air and water loops. Loops mimic the network of pipes and ducts found in real buildings and eventually will simulate head and thermal losses that occur as fluid moves in each loop. As mentioned earlier, EnergyPlus has no hardwired "template" systems. Instead, we developed input file templates for the each of the major system types in BLAST and DOE–2. These templates provide an easy starting point for users with system configurations that differ from "default" configurations. The air loop simulates air transport, conditioning, and mixing and includes supply and return fans, central heating and cooling coils, heat recovery, and controls for supply air temperature and outside air economizer. The air loop connects to the zone through the zone equipment.

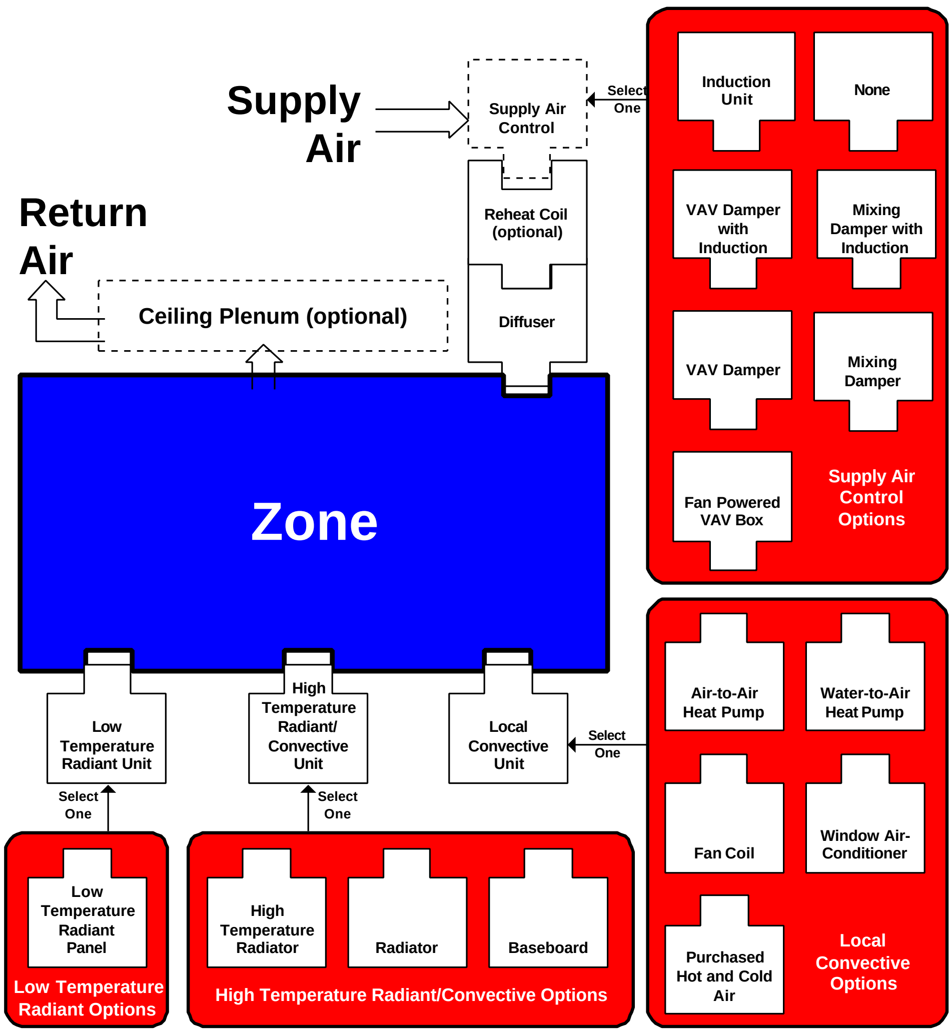

Figure 5: Zone Equipment Module

Zone equipment includes diffusers, reheat/recool coils, supply air control (mixing dampers, fan-powered VAV box, induction unit, VAV dampers), local convection units (window air-conditioning, fan coil, water-to-air heat pump, air-to-air heat pump), high temperature radiant/convective units (baseboard, radiators) and low temperature radiant panels. Figure 5 shows equipment connections to zones—note that more than one equipment type can be specified for a zone. However, users must specify equipment in the order it will be used to meet zone heating and cooling demand.

For the air loop, the solution method is iterative, not single-pass as in DOE–2 and BLAST. In order to specify equipment connections to a loop, nodes are defined at key locations around the loop with each node assigned a unique numeric identifier as shown in Figure 6. Node identifiers store loop state variables and set-point information for that location in the loop. We use an iterative solution technique to solve for unknown state variables along with control equation representations. These representations connect the set points at one node with the control function of a component, such as fan damper position and cooling coil water flow rate. In this schema, all the loop components are simulated first, then the control equations are updated using explicit finite difference. This procedure continues until the simulation converges. Typical control schemes are included in the input file templates described earlier.

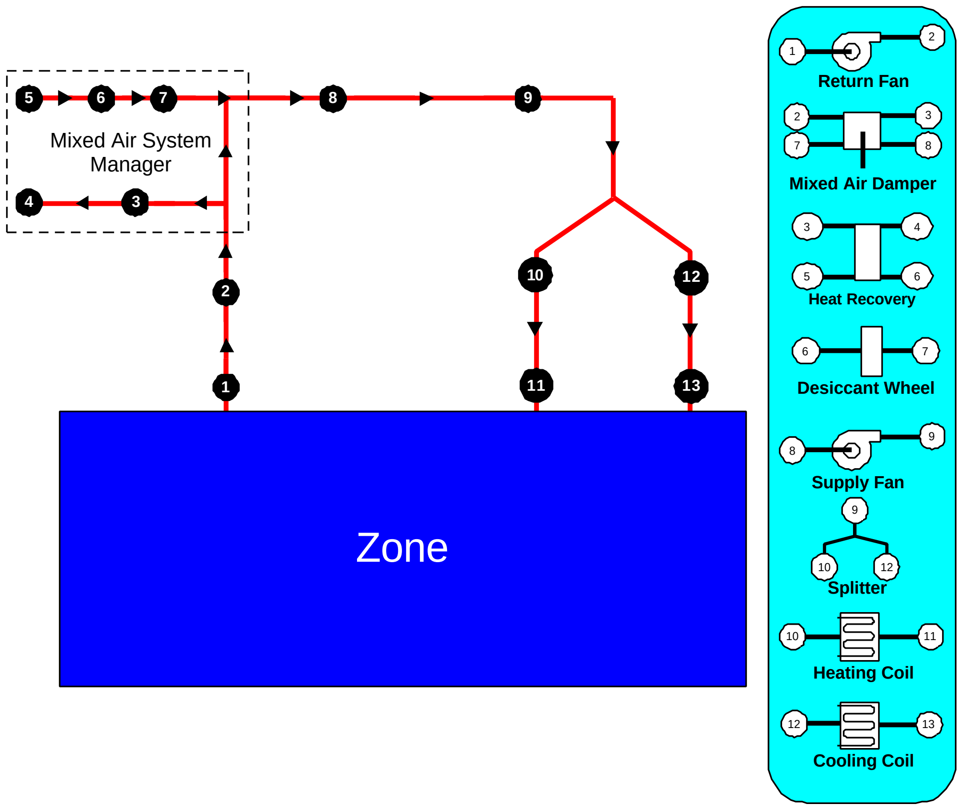

Figure 6: **Simple Air Loop Node Diagram**

There are two loops for HVAC plant equipment—a primary loop (for supply equipment such as boilers, chillers, thermal storage, and heat pumps) and a secondary loop (for heat rejection equipment such as cooling towers and condensers). Figure 7 presents a schematic view of equipment connections on the primary plant loop. Equipment is specified by type (gas-fired boiler, open drive centrifugal chiller) and its operating characteristics. In the first release of EnergyPlus, we are supporting performance-based equipment models (such as in BLAST and DOE–2). But because of the modular code, it will be easy for developers to add other types of models. As in the air loop, the primary and secondary plant loops use explicit nodes to connect equipment to each loop. Connections between the air loop and zone equipment and the primary and secondary loops are made through the node data structure and must be explicitly defined in the input file.

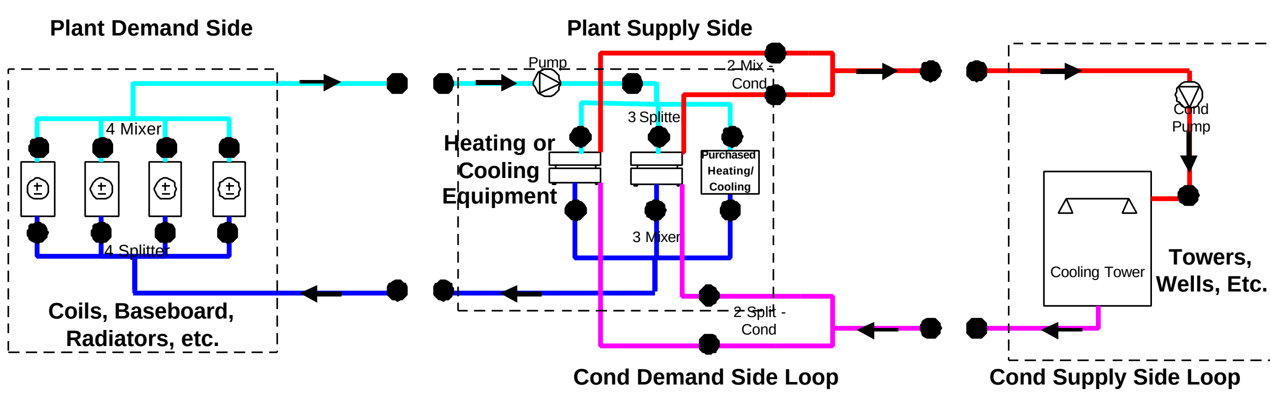

Figure 7: **Example Plant Loop**

A similar loop approach is proposed for a new electrical loop for simulating electrical systems—supply (utility, photovoltaic modules, and fuel cells), demand (plug loads, lighting, and other electrical loads), and measurement (meters).

In the longer term, EnergyPlus users will have more systems and equipment options through a link to SPARK [BUH 93], a new equation-based simulation tool. SPARK is a better solver for complex iterative problems and is currently in beta testing. SPARK already has a library of HVAC components based on the ASHRAE primary and secondary toolkits. EnergyPlus will continue to have system types (in input file templates) but developers and advanced users will be able to easily build complex new HVAC models with SPARK.

Input, Output, and Weather Data

Both DOE–2 and BLAST have highly-structured but user readable input file definitions that have evolved over many years. Instead of user readability, we designed the EnergyPlus input data files for easy maintenance and expansion. We chose to keep the input file simple in order to accept simulation input data from other sources such as CADD systems, programs that also do other functions, and pre-processors similar to those written for BLAST and DOE–2. An EnergyPlus input file, while readable, is cryptic and definitely not user-friendly—it is not intended to be the main interface for typical end-users. We expect most users will use EnergyPlus through an interface from a third-party developer. To make it easy for current DOE–2 and BLAST users to move to EnergyPlus, the team has written utilities that convert BLAST and DOE–2 input to the new EnergyPlus input structure.

Depending on how quickly the International Alliance for Interoperability (IAI) progresses in defining a de facto standard for building information exchange, a common object-oriented data store such as the IAI’s proposed Industry Foundation Classes [BAZ 97] may eventually become the main interface to the program.

EnergyPlus uses a free format input file that contains a complete object-based description of the building and its systems. The basic syntax is:

```
object, name, value, value, . . ., value;
```

"Object" is a pre-defined word denoting a building component, such as WALL, MATERIAL, LIGHTING, SYSTEM, HEATING COIL, and BOILER. This word is followed by a list of data values and terminates with a semicolon. These data describe performance characteristics and intended use for that object in the simulation. Unlike BLAST and DOE–2, the input file must explicitly provide all information—there are no default assumptions. Users may include comments throughout their input data file. A comparison of input file syntax for BLAST, DOE–2, and EnergyPlus is shown in Table 4.

During a simulation, EnergyPlus saves results for each time step in an output data structure. HVAC results are aggregated and reported at the time step. This structure uses a similar philosophy to the input—simple text files with a syntax of object, time stamp, data, data, data, . . ., data; . The output data is simple yet contains all the simulation results so that users and interface developers cam easily access specific results without modifying the calculation engine. Four types of reports are planned—standard output (aggregate hourly time step), one time output (such as input echo), detailed output (user-defined time step), and standard reports such as those in BLAST and DOE–2. Because the data structure is simple and comma-separated, output post-processors can easily read the data and create more elaborate reports. One drawback of our simple file format is that the output files can become very large.

The other major data input is weather. Rather than a binary file created by a separate weather processor, again we use a simple text-based format, similar to the input data and output data files. The weather data format includes basic location information in the first eight lines: location (name, state/province/region, country), data source, latitude, longitude, time zone, elevation, peak heating and cooling design conditions, holidays, daylight savings period, typical and extreme periods, two lines for comments, and period covered by the data. The data are also comma-separated and contain much of the same data in the TMY2 weather data set [NRE 95]. EnergyPlus does not require a full year or 8760 (or 8784) hours in its weather files. In fact, EnergyPlus allows and reads subsets of years and even sub-hourly (5 minute, 15 minute) data—the weather format includes a "minutes" field. EnergyPlus comes with a utility that reads standard weather service file types such as TD1440 and DATSAV2 and newer "typical year" weather files such as TMY2 and WYEC2.

In summary, all the data files associated with EnergyPlus—input, output, and weather—have simple self-contained formats but they can become quite large. The data files can be easily read and interpreted by other programs—spreadsheets, databases, or custom programs. By working with third party interface developers early, we will keep these files simple and easy to use by other programs that building designers use.

Table 4: Comparison of BLAST, DOE-2 (BDL) and EnergyPlus Input

		BLAST	BDL (DOE-2)	EnergyPlus
Location (Simple Input)	Description	Location defined in library. Library includes name, latitude, longitude, elevation, and time zone.	Location information defined by input, defaults to information on weather file	Location information defined by input.
	Input Syntax	LOCATION = Name;	BUILDING-LOCATION Latitude = W, Longitude = X, Altitude = Y, Time-Zone = Z ..	LOCATION, Name, Latitude, Longitude, Elevation, TimeZone;
	Example Input	**LOCATION = CHICAGO;**	**BUILDING-LOCATION LATITUDE = 41.98 LONGITUDE = 87.90 ALTITUDE = 673 TIME-ZONE = 6 ..**	**LOCATION, Chicago Illinois USA, 41.98, 87.90, 205, -6;**
Material (More Complex)	Description	Material defined in library. Library includes material name, conductivity, density, specific heat, resistance, roughness, and moisture properties.	Material from library or defined in input, includes thickness, conductivity, density, specific heat, or resistance. Thickness restated during Layer input (optional).	All material information defined by input.
	Input Syntax	TEMPORARY MATERIAL: Usname = (L=usn1, K=usn2, CP=usn3, D=usn4, ABS=usn5, TABS=usn6, R=usn7, TRANS=usn8, IR=usn9, FILMTRANS=usn10, REF=usn11, SC=usn12, roughness,asg); END;	A = Material, Thickness = W, Conductivity = X, Density = Y, Specific Heat = Z ..	MATERIAL, Name, Thickness, Conductivity, Density, Specific Heat, Roughness, Moisture Permeance, Moisture Resistance;
	Example Input	**Brick = (L=0.3333, K=5.6, CP=0.19, D=120, ROUGH);**	**BRICK = MATERIAL THICKNESS = 0.3333 CONDUCTIVITY = 5.6 DENSITY = 120 SPECIFIC-HEAT = 0.19 ..**	**MATERIAL, Brick, 0.1016, 0.721, 1922, 837, 46, 0.022;**

Adding A New Module

One of the main goals for EnergyPlus is to make it easy for developers to add new features and modules. The process is relatively simple. First, a developer defines a new module with model parameters and equations, specialized coefficients, and data needed. A developer then finds the "plug-in" point—where the module would be called within EnergyPlus. Next the developer writes the module (using the EnergyPlus programming standard), breaking the simulation tasks into modules. Finally, the developer writes new input file syntax based on the input needed for the module and uses EnergyPlus "get" routines to read the needed input data into the new simulation module. The input file syntax is not hardwired within EnergyPlus; instead EnergyPlus reads an input data dictionary at runtime to determine the syntax of the input data file. The general syntax is:

```
Object, A1 [what this is], N1 [a number],...;
```

For example, for the EnergyPlus Location command, the data dictionary line is:

```
Location, A1 [Location Name], N1 [Latitude], N2 [Longitude], N3
[Elevation], N4 [Time Zone];
```

This tells the input processor that, for the Location command, to expect one text field (A1) with the location name, and four numeric inputs (N1, N2, N3, and N4)–latitude, longitude, elevation, and time zone respectively. Words in brackets [ ] describe the variable and its units (meters, liters/second, etc.).

Release 1.0 and Beyond

"More people have ascended bodily into heaven than have shipped great software on time." [McC 95]

The first working version of EnergyPlus, or alpha version, was completed in December 1998 for internal testing by the team. The alpha version did not contain all the modules intended for the first release of EnergyPlus—those will be included in the first beta version, an internal version for testing that will be completed in Spring 1999. By Summer 1999, a beta version will be available to outside developers for testing. Shortly thereafter, a beta test version will be available for general testing. We plan to release version 1.0 of EnergyPlus in early 2000.

In late 1999 we will begin planning for the second release of EnergyPlus based on new features suggested by users, developers, and the team. Working with a coordinating group of users and developers, we will select the features and capabilities for that release. We plan to release updates to EnergyPlus on an 18-month release cycle. Some new features already under development are a connection to the COMIS airflow program, improved ground heat transfer, electrical system simulation, solar thermal and photovoltaic modules, and link to SPARK. The link to COMIS will allow better calculation of infiltration, natural ventilation, multi-zone airflow, and air pollutant transport. The ground heat transfer model will either be a 2-dimensional or 3-dimensional heat transfer calculation for various foundation calculations.

Summary

EnergyPlus is a new building energy simulation program that builds on the strengths of BLAST and DOE–2. It is being written in Fortran 90 with structured, modular code that is easy to maintain, update, and extend. Benefits of EnergyPlus include:

For simulation program users:

- limits built into BLAST and DOE–2 (such as number of zones, schedules, or systems) are eliminated by the new structures in EnergyPlus—now limited only by a user's computer resources rather than hardwired in code
- EnergyPlus source code is open for inspection—and understandable
- developers around the world will be able to develop new modules—algorithmic or interfaces new module development can keep pace with new building technologies, maximizing public impact of latest buildings research

For simulation developers:

- standardized structure significantly decreases the learning curve for developers
- new, structured, modular code is easier to understand and work with
- modular structure allows developers to work in parallel on new modules

General benefits include:

- simulation capabilities include integrated simulation, combined heat and mass transfer balance, multi-zone airflow, HVAC loops (flexible system and plant simulation), links to SPARK system/plant simulation, and algorithms from the new ASHRAE Loads Toolkit
- input, output, and simulation capabilities are much more flexible
- EnergyPlus will be released quicker than a next-generation program, but offer similar benefits

Although the two workshops sponsored by DOE and DOD [CRA 97] pointed up the critical need for good user interfaces in the success of any simulation tool, the EnergyPlus team is focusing first on developing the heart of a new simulation tool—the calculation engine. We consciously incorporated the priorities of the workshop participants in our development effort (many can be seen in Figure 1). The EnergyPlus team has begun working with third party interface developers to ensure user-friendly interfaces and new modules are ready when the program is released.

EnergyPlus not only combines the best features of the BLAST and DOE–2 programs, but also represents a significant step towards next-generation building simulation programs both in terms of computational techniques and program structures. Connectivity and extensibility are overriding objectives in the design and development process. This will ensure broad participation in program enhancement and facilitate third party interface and module development. EnergyPlus beta testing begins in early 1999. Up to date information on EnergyPlus is provided on the EnergyPlus web site.

Web Resources

1. Information on EnergyPlus including schedule, documentation, programming standards, and availability of beta releases: **www.eren.doe.gov/buildings/energy_tools/energyplus.htm**

2. Workshops on Next Generation Energy Simulation Tools: **www.eren.doe.gov/buildings/energy_tools/workshops.htm**

3. Web-based directory of more than 130 building-related software tools from around the world: **www.eren.doe.gov/buildings/tools_directory/**

Acknowledgements

____

____

____

____

____

____

____

____

References

ARA 94 Arasteh, D. K., E. U. Finlayson, and C. Huizenga. 1994. *Window 4.1: Program Description*, Lawrence Berkeley National Laboratory, report no LBL-35298. Berkeley, California: Lawrence Berkeley Laboratory.

BAZ 97 Bazjanac, Vladimir, and Drury B. Crawley. 1997. "International Alliance for Interoperability: The Implementation of Industry Foundation Classes in Simulation Tools for the Building Industry," in *Proceedings of Building Simulation '97*, Volume I, pp.203-210, September 1997, Prague, Czech Republic, IBPSA.

BLA 92 BLAST Support Office. 1992. *BLAST 3.0 Users Manual*. Urbana-Champaign, Illinois: BLAST Support Office, Department of Mechanical and Industrial Engineering, University of Illinois.

BUH 93 Buhl, W.F., A.E. Erdem, F. C. Winkelmann and E. F. Sowell. 1993. "Recent Improvements in SPARK: Strong-Component Decomposition, Multivalued Objects and Graphical Editors," in *Proceedings of Building Simulation '93*, pp. 283-289, August 1993, Adelaide, South Australia, Australia, IBPSA.

CRA 97 Crawley, D. B., L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, A. E. Erdem, C. O. Pedersen, R. J. Liesen, and D. E. Fisher. 1997. "The Next-Generation in Building Energy Simulation—A Glimpse of the Future," in *Proceedings of Building Simulation '97*, Volume II, pp. 395-402, September 1997, Prague, Czech Republic, IBPSA.

FEU 90 Feustel, H. E. 1990. "The COMIS Air-Flow Model – A Tool for Multi-zone Applications," in *Proceedings of the 5th International Conference on Indoor Air Quality and Climate*, Vol. 4, pp. 121-126.

McC 95 McCarthy, Jim. 1995. *Dynamics of Software Development*, pg. 2. Redmond, Washington: Microsoft Press.

MET 95 Metcalf, R. R., R. D. Taylor, C. E. Pedersen, R. J. Liesen, and D. E. Fisher. 1995. "Incorporating a Modular System Simulation Program into a Large Energy Analysis Program: the Linking of IBLAST and HVACSIM+, " in *Proceedings of Building Simulation '95*, pp. 415-422, August 1995, Madison, WI, IBPSA.

NRE 95 National Renewable Energy Laboratory (NREL). 1995. *User's Manual for TMY2s (Typical Meteorological Years)*, NREL/SP-463-7668, and TMY2s, Typical Meteorological Years Derived from the 1961-1990 National Solar Radiation Data Base, June 1995, CD-ROM. Golden, Colorado: NREL.

TAY 90 Taylor, R. D, C. E. Pedersen, and L. K. Lawrie. 1990. "Simultaneous Simulation of Buildings and Mechanical Systems in Heat Balance Based Energy Analysis Programs," in *Proceedings of the 3rd International Conference on System Simulation in Buildings*, Liege, Belgium, December 3-5, 1990.

TAY 91 Taylor R. D., C. E. Pedersen, D. E. Fisher, R. J. Liesen, and L. K. Lawrie. 1991. "Impact of Simultaneous Simulation of Building and Mechanical Systems in Heat Balance Based Energy Analysis Programs on System Response and Control," in *Proceedings of Building Simulation '91*, August 1991, Nice, France.

WIN 85 Winkelmann, F. C. and S. E. Selkowitz. 1985. "Daylighting Simulation in the DOE–2 Building Energy Analysis Program," in *Energy and Buildings*, 8, pp. 271-286.

WIN 93 Winkelmann, F. C., B. E. Birdsall, W. F. Buhl, K. L. Ellington, A. E. Erdem, J. J. Hirsch, and S. Gates. 1993. *DOE–2 Supplement, Version 2.1E*, LBL-34947, November 1993, Lawrence Berkeley National Laboratory. Springfield, Virginia: National Technical Information Service.