STADIC - Full Building Daylight Modeling

Richard Mistrick, Penn State

Craig Casey, Penn State & Lutron Electronics

Sarith Subramaniam, Penn State

Overview

- This presentation introduces the Penn State STADIC library (Simulation Tool for Architectural Daylight and Integrated Control) for conducting annual daylight simulations using Radiance.
- A key element of this library is the application of a JSON control file that contains all of the necessary input data related to these simulations.
- These programs are in the final stages of development and will be distributed as a separate library along with Radiance. The code is open source and released under the Radiance license.

Rules of Engagement

- Feedback on the usefulness and utility of the STADIC programs is requested.
- Suggestions of additional features and capabilities are strongly encouraged.

Background

- Annual simulations will soon be the norm.
- Codes and rating systems (such as LEED) are beginning to require or encourage annual performance modeling of energy as well as the evaluation of recently developed annual daylight metrics (DA, sDA, ASE, etc.).
- Many spaces include complex geometry and, in most cases, require simulation and operation of shades or blinds (i.e., complex fenestration systems, CFS).
- BSDF's simplify modeling of complex fenestration.
- Summary - Simulations are getting complex and new tools are necessary.

Recent Radiance Developments

- rfluxmtx
- gendaymtx
- rcontrib
- 3-phase and 5-phase simulations with BSDF's
- . . . and others

Putting these all together for a real project can become complicated and time consuming.

Penn State's History with Annual Simulations

- Penn State added the ability to simulate electric lighting systems, shade geometry, and photocontrol along with daylighting in DAYSIMps. www.daysim.ning.com
- DAYSMps utilizes a Javascript interface, with some calculations performed within the GUI.
- DAYSIM (and DAYSIMps) utilize an ASCII header file with select keywords for inputs to the analysis for a single space.
- With support from DOE and CBEI (and assistance from NREL and LBL), C++ programs were created to process annual simulations and metrics, and to model photocontrol, for integration into other applications (such as OpenStudio).
- Object-oriented programming was a prime focus in the development of these programs.
- A versatile input data format was created.

STADIC Development

- Goal – To generate a daylight simulation manager for Radiance, and an accompanying input data file structure, that serve both Radiance users and software developers.
- Simulations should apply only standard Radiance binaries.
- Input data is extensive. What file structure should be used to describe the following?
  - A building's collection of spaces
  - The electric lighting in each space
  - The daylight elements, specified as window groups, with their shading elements
  - Control of the shades and the electric lighting
  - The simulations to be performed
- A JSON formatted input file was selected.

Design Issues

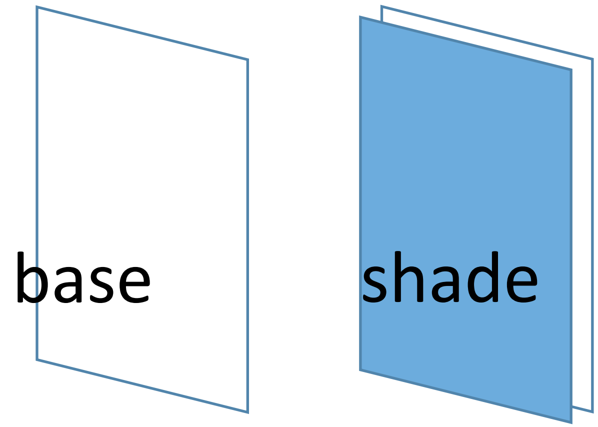
- Daylight simulations in the future are likely to be required for a large portion of a building.
- Space level simulations and data are required to control shades and electric lighting on a space-by-space basis.
- CAD/BIM models can export space data (A separate Penn State team is working on this).
- Simulation formats are likely to vary at different stages of the building design process. Flexibility is essential.
- The input files should be readable.
- Parallel processing, BSDFs, etc., should be utilized to maximize performance and minimize simulation time.

PennState CBEI CONSORTIUM for BUILDING ENERGY INNOVATION LUTRON

Materials add to the complexity

STADICS's dxDaylight program is designed to handle multiple material configurations in its models.

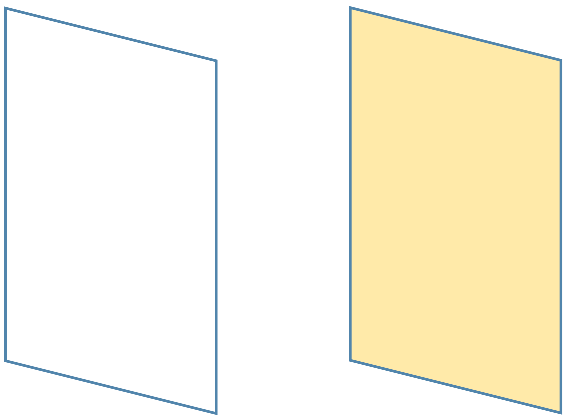
- 1 Standard materials only – no BSDFs.



Materials add to the complexity

STADICS's dxDaylight program is designed to handle multiple material configurations in its models.

- 1 Standard materials only – no BSDFs.
- 2 Standard materials in base case, exchanged with BSDFs in shade settings.

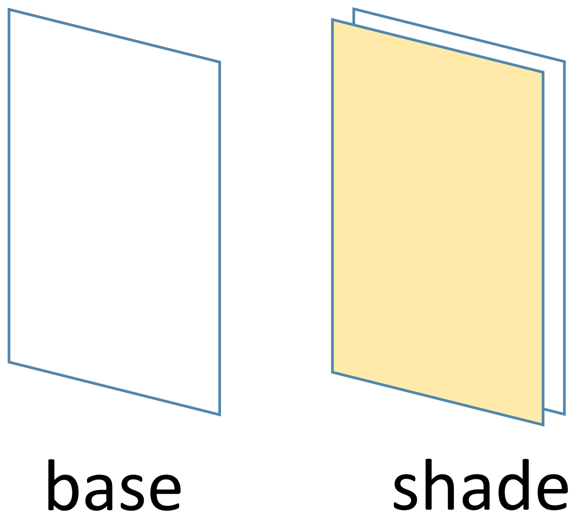


base shade

Materials add to the complexity

STADICS's dxDaylight program is designed to handle multiple material configurations in its models.

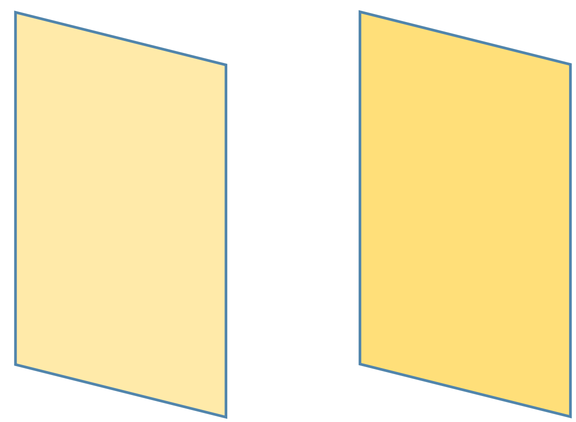
- 1 Standard materials only – no BSDFs.
- 2 Standard materials in base case, exchanged with BSDFs in shade settings.
- 3 Standard materials in base case, with BSDFs added in shade settings.



Materials add to the complexity

STADICS's dxDaylight program is designed to handle multiple material configurations in its models.

- 1 Standard materials only – no BSDFs.
- 2 Standard materials in base case, exchanged with BSDFs in shade settings.
- 3 Standard materials in base case, with BSDFs added in shade settings.
- 4 BSDFs in base case, exchanged with other BSDFs.

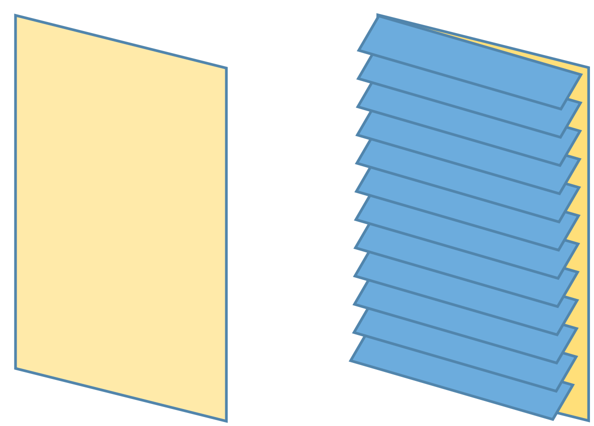


base shade

Materials add to the complexity

STADICS's dxDaylight program is designed to handle multiple material configurations in its models.

- 1 Standard materials only – no BSDFs.
- 2 Standard materials in base case, exchanged with BSDFs in shade settings.
- 3 Standard materials in base case, with BSDFs added in shade settings.
- 4 BSDFs in base case, exchanged with other BSDFs.
- 5 BSDFs in base case, exchanged with other BSDFs - with proxy geometry included.



base   shade

PennState CBEI CONSORTIUM for BUILDING ENERGY INNOVATION LUTRON

How Complex can a simulation get?

- For a room with two window groups, and two shade settings on each, and non-cosine photocontrol of the shades, how many simulations are needed?
- Basic Model (no BSDF's) – using sky and sun DC's
  - Base case illuminances, window group 1
  - Base case illuminances, window group 2
  - Illuminances for shade setting 1 on window group 1
  - Illuminances for shade setting 2 on window group 1
  - Illuminances for shade setting 1 on window group 2
  - Illuminances for shade setting 2 on window group 2

  Plus
  - Photocontrol signals for window group 1
  - Photocontrol signals for window group 2

. . . .continued

- Basic Model (no BSDF's) – using sky and sun DC's

  If electric light is added with one control zone and a photosensor,

  - Illuminances from each lighting zone
  - Signals from each lighting zone
  - Signals for base case, window group 1
  - Signals for base case, window group 2
  - Signals for shade setting 1 on window group 1
  - Signals for shade setting 2 on window group 1
  - Signals for shade setting 1 on window group 2
  - Signals for shade setting 2 on window group 2

BSDF data make this a bit easier

- View matrix to window group 1 (full, sky patches, & direct)
- View matrix to window group 2 (full, sky patches, & direct)
- Daylight matrix to window group 1
- Daylight matrix to window group 2
- Group 1 sensor view matrix (full, sky patches, & direct)
- Group 2 sensor view matrix (full, sky patches, & direct)
- Electric light photosensor view matrix to window group 1
- Electric light photosensor view matrix to window group 2
- Sky matrices
- Final solution for each (assuming five-phase):

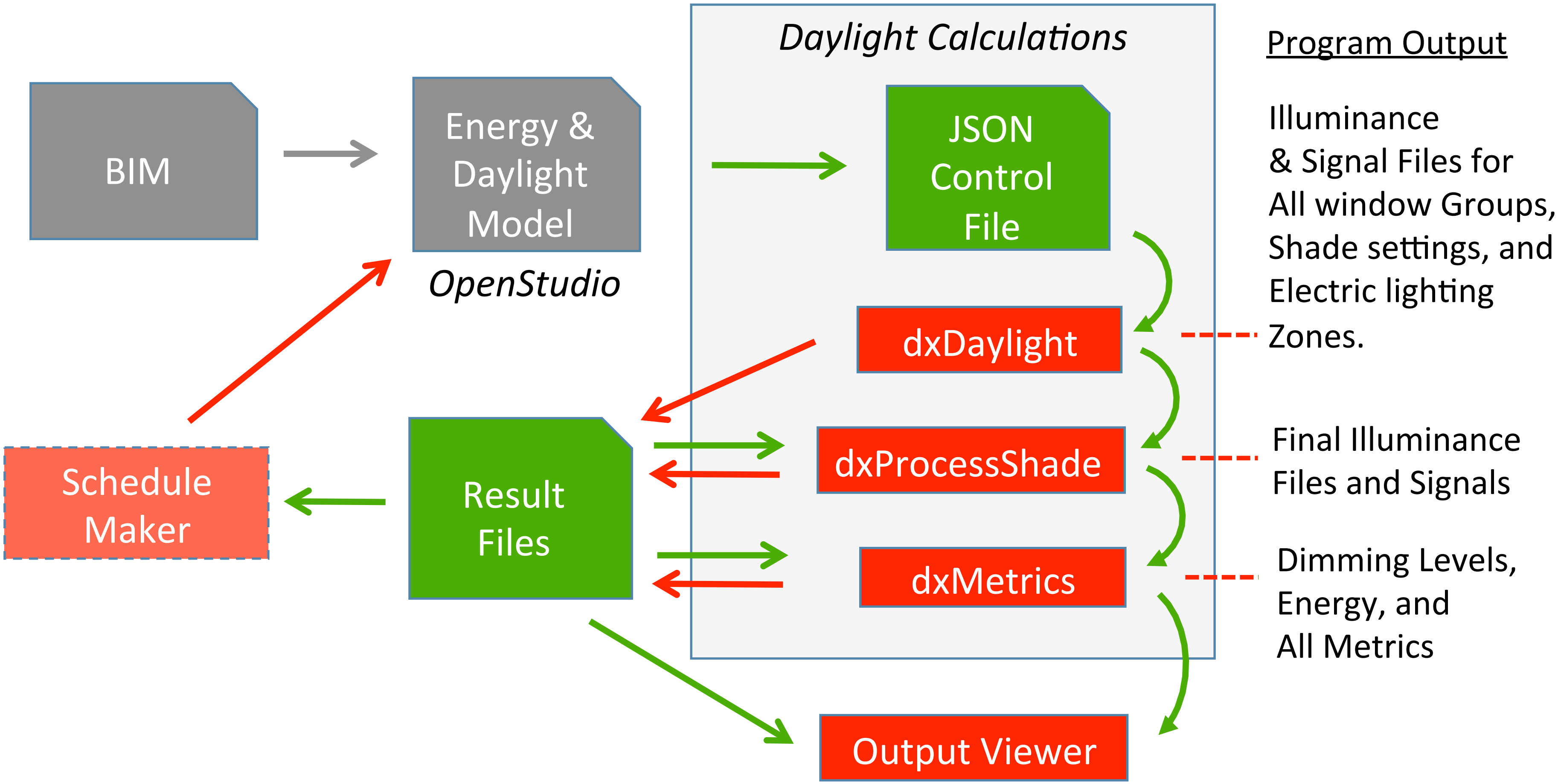
$$I_{5ph} = VTDS - V_d T D_d S_d + C_{ds} S_{sun}$$

PennState CBEI CONSORTIUM for BUILDING ENERGY INNOVATION LUTRON

Additional Concern with BSDFs

- With BSDF data and exterior shading, it may be necessary to discretize a window group into multiple polygons, each with their own coefficients.

STADIC's Daylight Simulation Path



PennState CBEI CONSORTIUM for BUILDING ENERGY INNOVATION LUTRON

STADIC's Daylight Simulation Path

- Running STADIC consists of three simple commands:

```
dxdaylight BLDGfile.JSON
dxprocessshade BLDGfile.JSON
dxmetrics BLDGfile.JSON
```

The output utility is then used to view the results.

JSON Overview - Basic Structure

- Keywords are used to describe the data item on each line. Keywords are placed in quotes and are followed by a colon (:). Items in a list end in a comma.

  Example: **"base_geometry": "W-1-base.rad",**

- [ ]'s contain <u>data</u> to be listed in **arrays** – all data listed within [ ]'s contain the same information, such as a listing of different shade settings that might be applied. Items are separated by a comma and there is no comma after the last item.

  Example: **"angle_settings" : [20, 40, 60]**

- { }'s contain a list of different <u>data items</u> that describe an **object**. Items are separated by a comma, with no comma after the last item.

  Example: **"DA": {"calculate": false, "illuminance": 300}**

- Line feeds are ignored.

```
"DA": {
        "calculate": false ,
        "illuminance": 300
}
```

A JSON Example for window groups

Red = Arrays

Blue = Objects

```
"window_groups": [
        {
            "name": "WINDOWS-1",
            "base_geometry": “W-1-base.rad",
            "calculate_base": true,

            "glazing_materials": [
              "l_glass"
            ],

            "shade_control": {
              "method":
"automated_profile_angle",
              "elevation_azimuth": 0,
              "angle_settings" : [40]
             },

            "shade_settings": [
              “W-1-SET1.rad"
            ],

            "calculate_setting": [
              true
            ]

        }
    ],
```

STADIC's JSON Control File - Content

- General & default information related to the model (applies to all spaces, unless overridden).
- List of spaces with all relevant data
  - Space name
  - Folders/directories for input and output
  - Schedule(s)
  - .rad files
  - Analysis points (file or specifications for dxGridmaker)
  - Window group specifications, with transmissive material listed, shading conditions, and control information.
  - Calculation switches

STADIC's JSON Control File - Content

- Electric lighting system control zones
  - Name
  - Type of control with settings
  - Luminaire data
  - Ballast/driver data
  - Luminaire locations
- Calculations desired
  - DF, DA, cDA, UDI, sDA, ASE
  - (Dimming Levels and Energy are computed based on space control settings)

STADIC's Calculations

- Numerous Shading Control Options are available
- Electric lighting control to evaluate potential lighting energy savings can also follow several different paths.

- New options of each can be easily added in the future.

Available Options – Shade Operation

- sDA Control – Shade closure on window groups is optimized to limit direct sunlight to no more than 2% of the space at 1000 lux or higher.
- Profile Angle – shade settings are based on the solar profile angle on that façade, regardless of the sky condition.
- Sensor Signal – shade settings are based on the reading of a user-specified photosensor or illuminance meter for the base window condition.
- Profile Angle + Sensor Signal – shade settings are based both the profile angle and a sensor signal at the base window condition.
- Glare –Similar to the algorithm presently used in OpenStudio/EnergyPlus .
- Others . . . . TBD

Available Options – Electric Lighting

- EnergyPlus mode – Lighting Power Density and percent of space controlled with user supplied reference points used to evaluate savings.
- Specified electric lighting system layouts
  - Option 1 - Optimum control based on a specified number of work plane critical points, which are determined by the simulation.
  - Option 2 - Analyzed via a specified photosensor sensitivity distribution, location, and control algorithm, applying an algorithm setting that limits overdimming to a specified fraction of the time.

Available Options – Electric Lighting

- Available Photocontrol Algorithms
  - Open-Loop Switching
  - Closed-Loop Switching
  - Closed or Open Loop Linear Proportional Dimming
  - Closed-Loop Constant Setpoint Dimming
  - Closed-Loop Custom Dimming Algorithm (a series of curves along a continuum)

How do Developers access the JSON file?

- The STADIC library contains parser objects (for building, shading, and electric light data) that can be incorporated into any software program that requires input data contained in the JSON control file.

PennState CBEI CONSORTIUM for BUILDING ENERGY INNOVATION LUTRON

Let’s Tour An Example STADIC Control File

Evaluating the Output

- The STADIC utilities generate a large amount of output that is best view interactively.
- The STADIC Output Viewer permits dynamic viewing and evaluation of the output, based on the analyses requested in a JSON control file.
- Single space evaluation and a full-building summary are provided.

Let's Look at the Operation of the Output Viewer

Summary

- STADIC's analysis programs and input data format are designed to make powerful and detailed daylight simulation processing easier.
- The design is flexible and expandable, and addresses the needs of both users of Radiance and other software tool developers.

We look forward to your comments and questions.

THANK YOU!