Radiance for Lighting Designers

A (non-)programmers perspective with pictures

Thomas Bleicher

BDP Lighting
What BDP Lighting does

Historic/Art: Almshouses

Public: National Gallery
What BDP Lighting does

Office: Roche

Retail: Breahead
What BDP Lighting does

Schools: lots
Hapden Gurney

... always an eye on the detail
What BDP Lighting does

Exterior: Cardinal Place

but also: car parks
case study: car park
case study: car park
case study: car park
What’s Radiance ...

... for the Lighting Designer?

- too !|>& complicated
- very technical (units, concepts)
- for special cases (like daylight)
- “miracle without price tag”
- slow
The Lighting Designer’s friends

AGI32, DIALux, Dulux, Pollux, Relux, ...

- accurate enough *(at least, we hope they are …)*
- supported by manufacturers
- specialised for artificial lighting
- simple *(but restricted)*
- integrated solutions (setup, calculation, visualisation, report, …)
- fast *(radiosity)*
- free, expensive - who cares?
example: DIALux

- financed, supported and promoted by the DIALux Consortium
- developed by DIAL AG, Germany
- radiosity based, closed source
- Windows only
- freeware
- luminaire support via files, plugins, online DBs, drag-n-drop
- POVRay as optional renderer
What DIALux does well

- quick setup of *simple* scenes
- luminaire layout options (line, grid, spacing)
- quick rendering of large scenes
- integration of standards (emergency etc.)
- 3D view with light distribution
- PDF reports (lots of details for luminaires)
- variable display of results (isolines, greyscale, falsecolour)
What DIALux does well
What DIALux does not so well

- interface
  - inflexible tree view
  - unintuitive and complicated
- implementation
  - grouping/editing of luminaires
  - variation/referencing of scene data
- data exchange
- compatibility with older versions
- user support and feedback
- DIALux Cabal
DIALux POVRay export

- simple polygon data (generated from cubes)
- textures and materials from predefined range
- no POVRay scripting
- contains distribution data
  
  ➡ left-handed coordinate system
  ➡ y-axis is ‘up’
  ➡ only one viewpoint
  ➡ no ‘extra’ information (sky!)
emergency snapshot of DIALux running POVray as external renderer
POVray to Blender: DIAray.py

perspective view in DIALux

perspective view in Blender
DIARay.py

omissions and roadmap

• rotation of luminaires
• skylights
• import luminaire data
  • identify different luminaires
  • use data from *.pov file
• colour

• Radiance server with web frontend
  ➡ Radiance for everyone
Blender to Radiance: b/rad 2.0
b/rad 2.0 roadmap

... or wishlist

• clean up and consolidate
• merge bliff and b/rad
• establish project structure (Rayfront)
• ultimate luminaire library
• Radiance scene import
• network libraries and rendering
• sync BlenderDummy functions
• Radiance library for Python