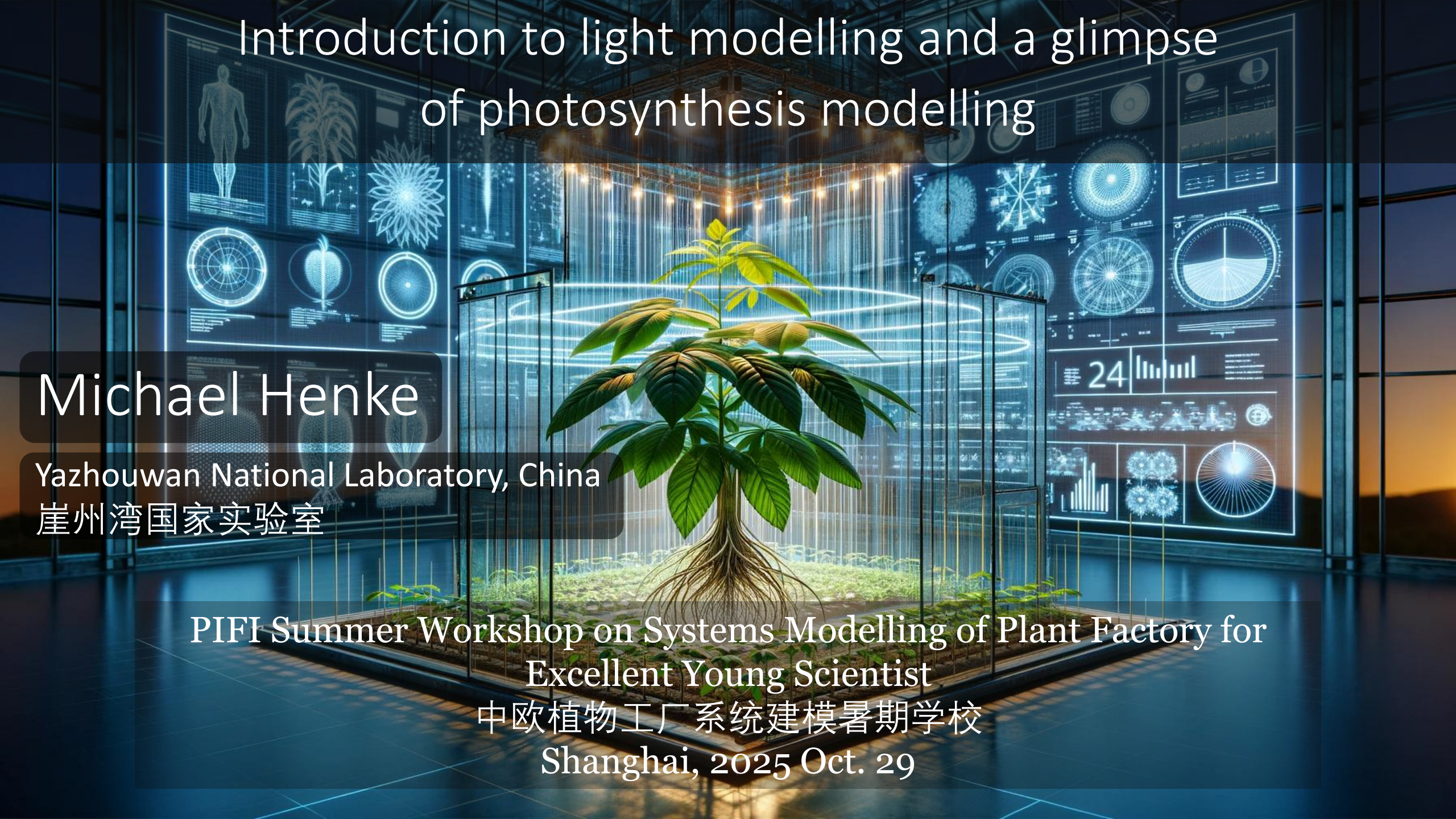
Introduction to light modelling and a glimpse of photosynthesis modelling

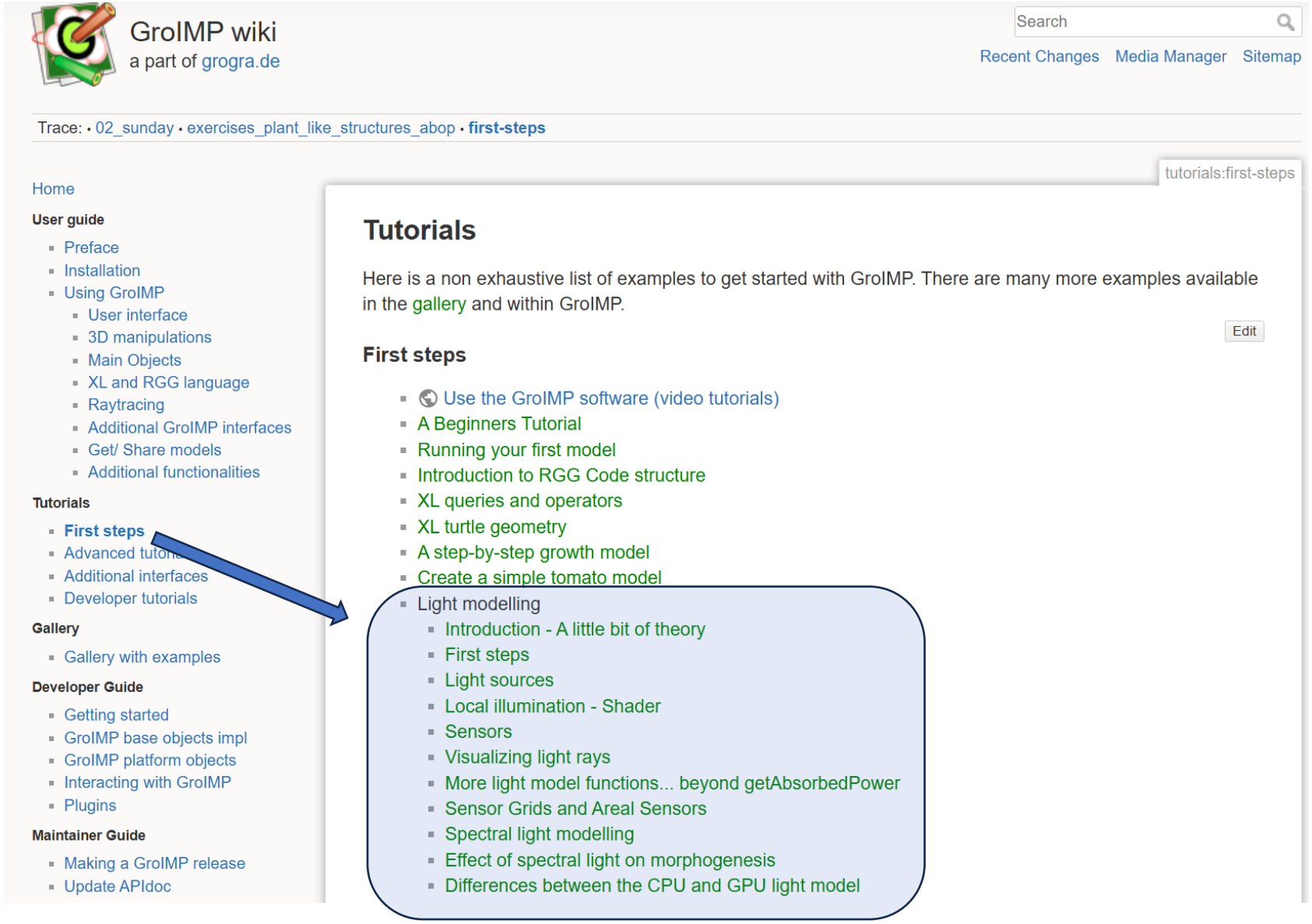


Michael Henke

Yazhouwan National Laboratory, China
崖州湾国家实验室

PIFI Summer Workshop on Systems Modelling of Plant Factory for Excellent Young Scientist
中欧植物工厂系统建模暑期学校
Shanghai, 2025 Oct. 29

Online Tutorials

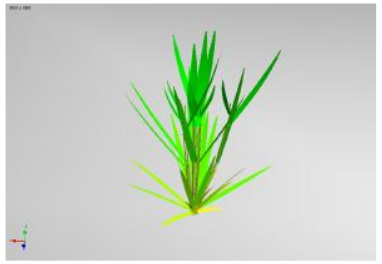


Wiki: https://wiki.grogra.de/doku.php?id=tutorials:first-steps

Light modelling – Definition

Light modelling is the numerical solution of how electromagnetic energy is **emitted, transported, scattered, absorbed, and measured** in a scene. Practically, it predicts **radiance** at objects from known **emitters**, **geometry**, and **materials**, across wavelength and time.

Input: 3d scene



Rendering vs Light modelling

rendering

light calculation

Output: image vs data

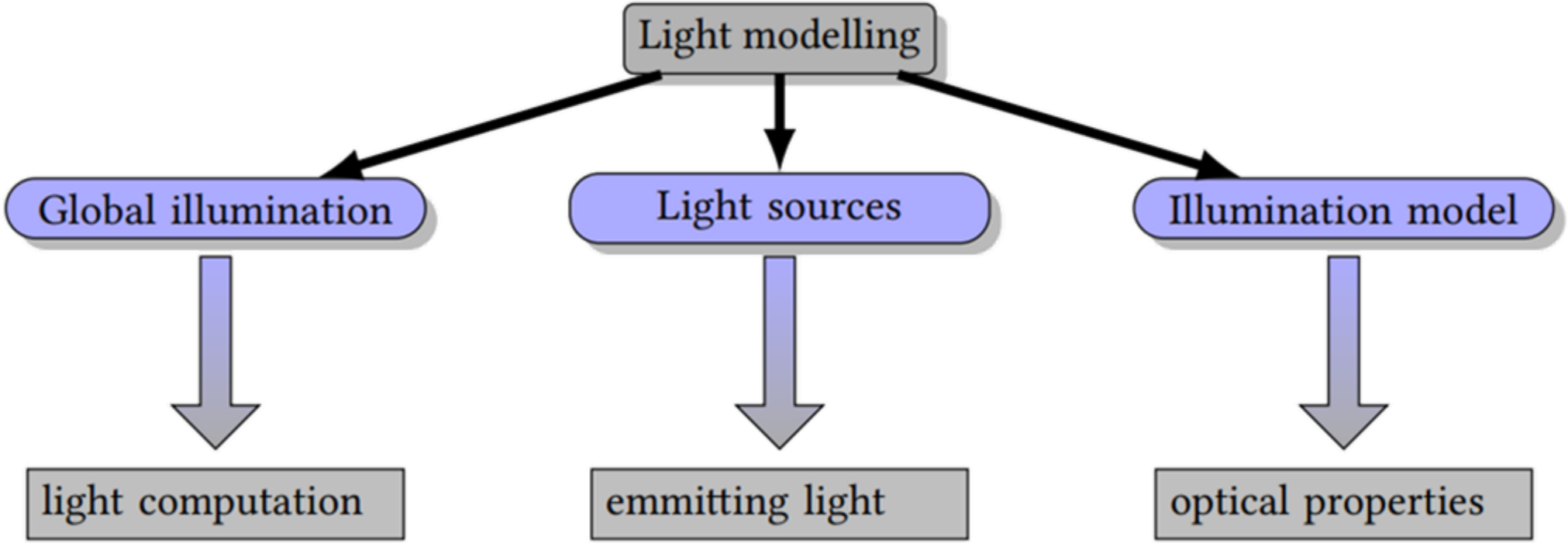
process of generating an image

1	0.8234641
2	0.7027052
3	1.3539617
4	0.6167236
5	0.6495002
6	1.3293481
...	

process to calculate the light distribution

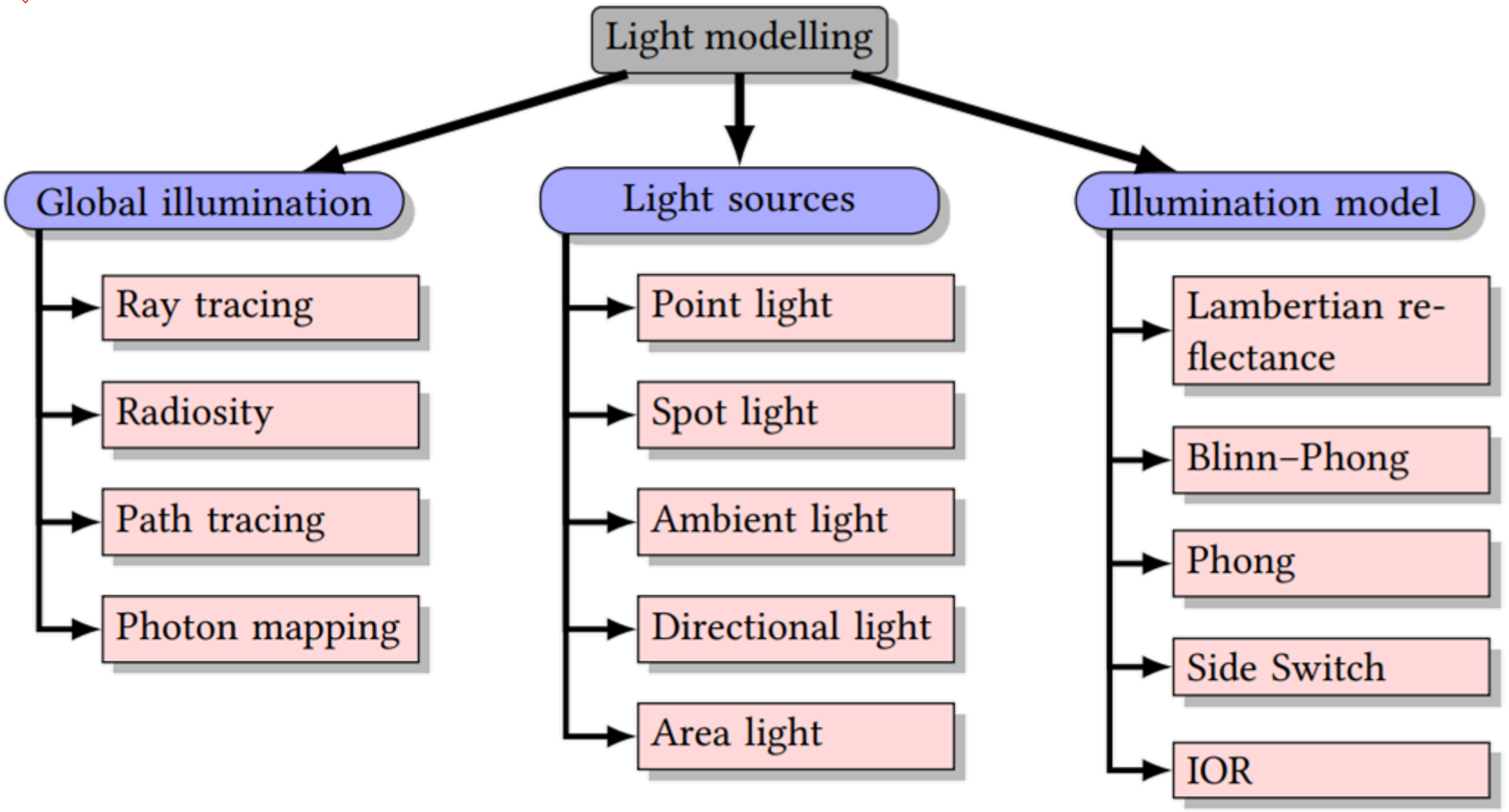
Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-introduction

Light modelling – Aspects



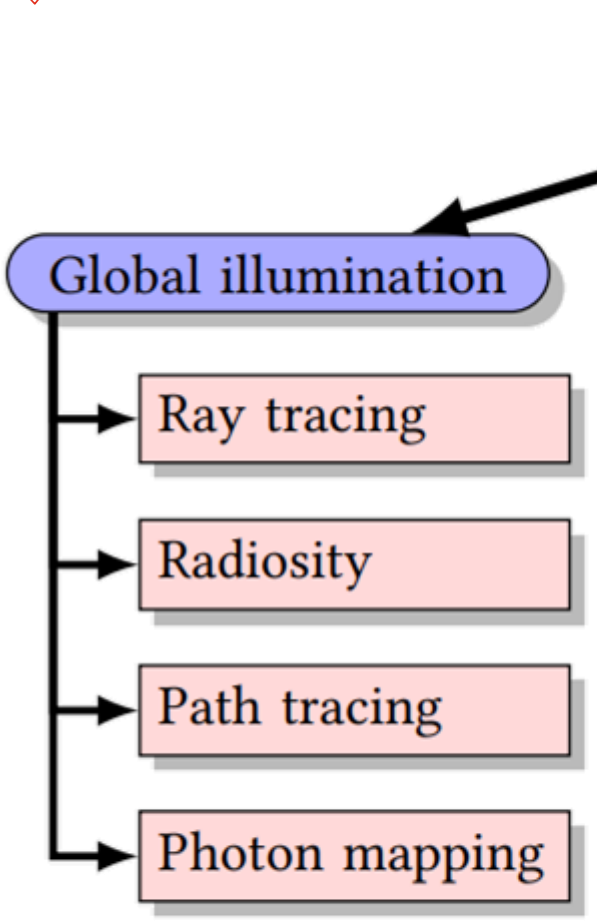
Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-introduction

Light modelling – Aspects



Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-introduction

Light modelling – Aspects: Global illumination models



GroIMP integrates two main light model implementations, namely:

- **Twilight, a CPU-based implementation**
- **GPUFlux, a GPU-based implementation**

Both implementing different global illumination models for rendering and for light computation.

Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-introduction

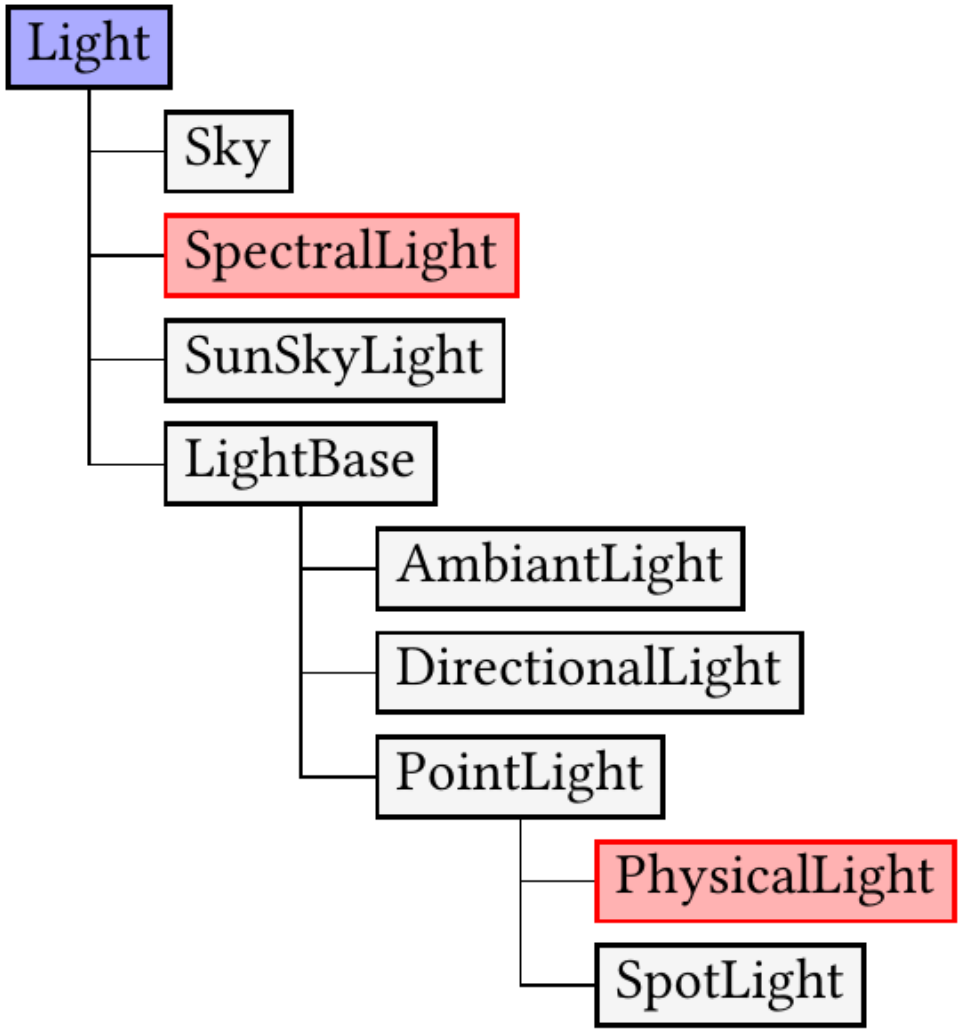
Light modelling – Aspects: Global illumination model


```
21  // initialize the light model
22  LightModel CPU_LM = new LightModel(RAYS, DEPTH);
23
24  CPU_LM.compute(); // run the light model
25
26  //check the scene objects for their light absorption
27  Spectrum ms;
28  [
29    x:Box::> { ms = CPU_LM.getAbsorbedPower(x); }
30  ]
31  print("absorbed = "+ms);println(" = "+ms.integrate(), 0xff0000);
```

Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-introduction

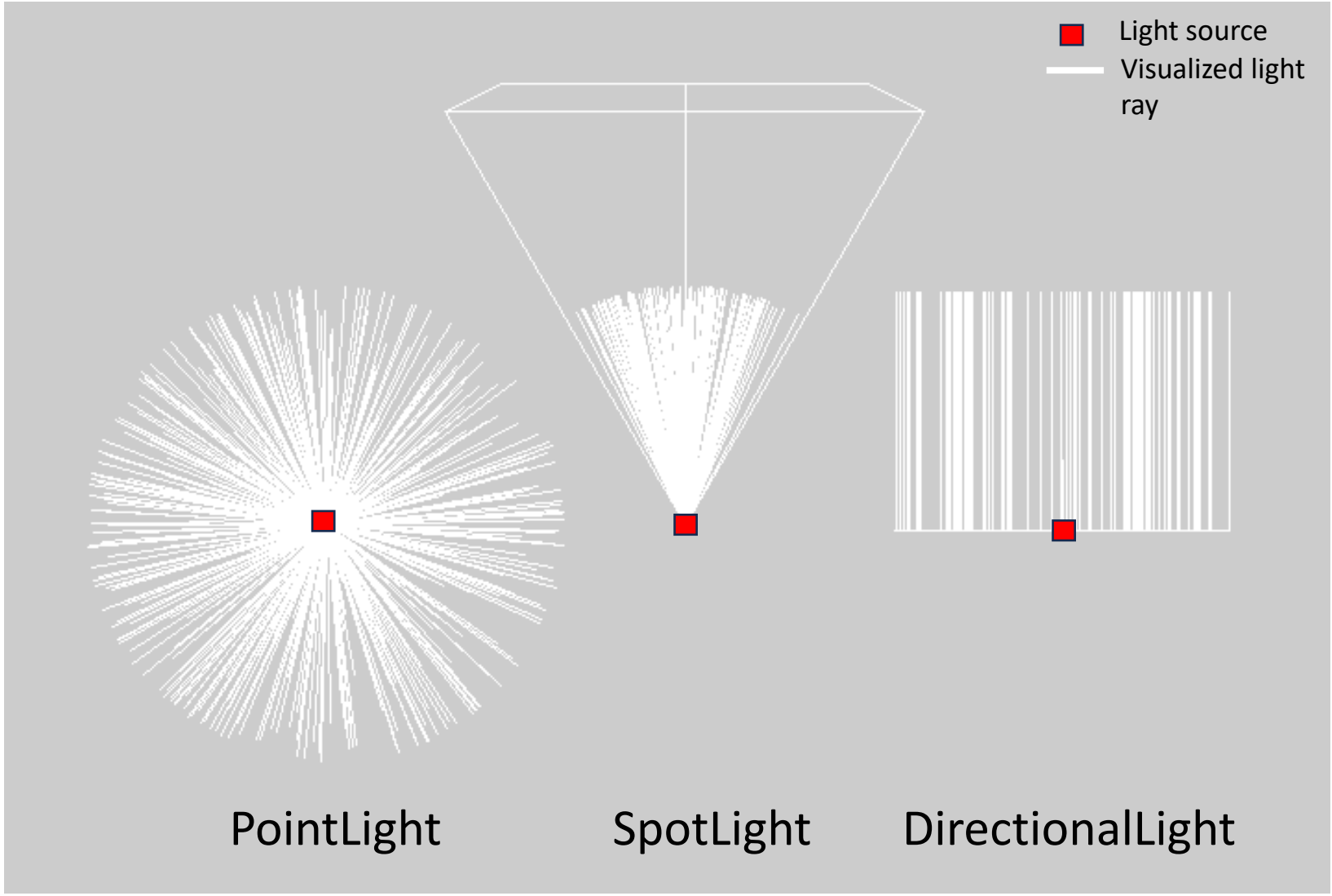
Light modelling – Aspects: Light sources

- GroIMP provides a complete set of possible implementations
- All implement the *Light* and *LightBase* interfaces, which makes them easy to handle and exchange.



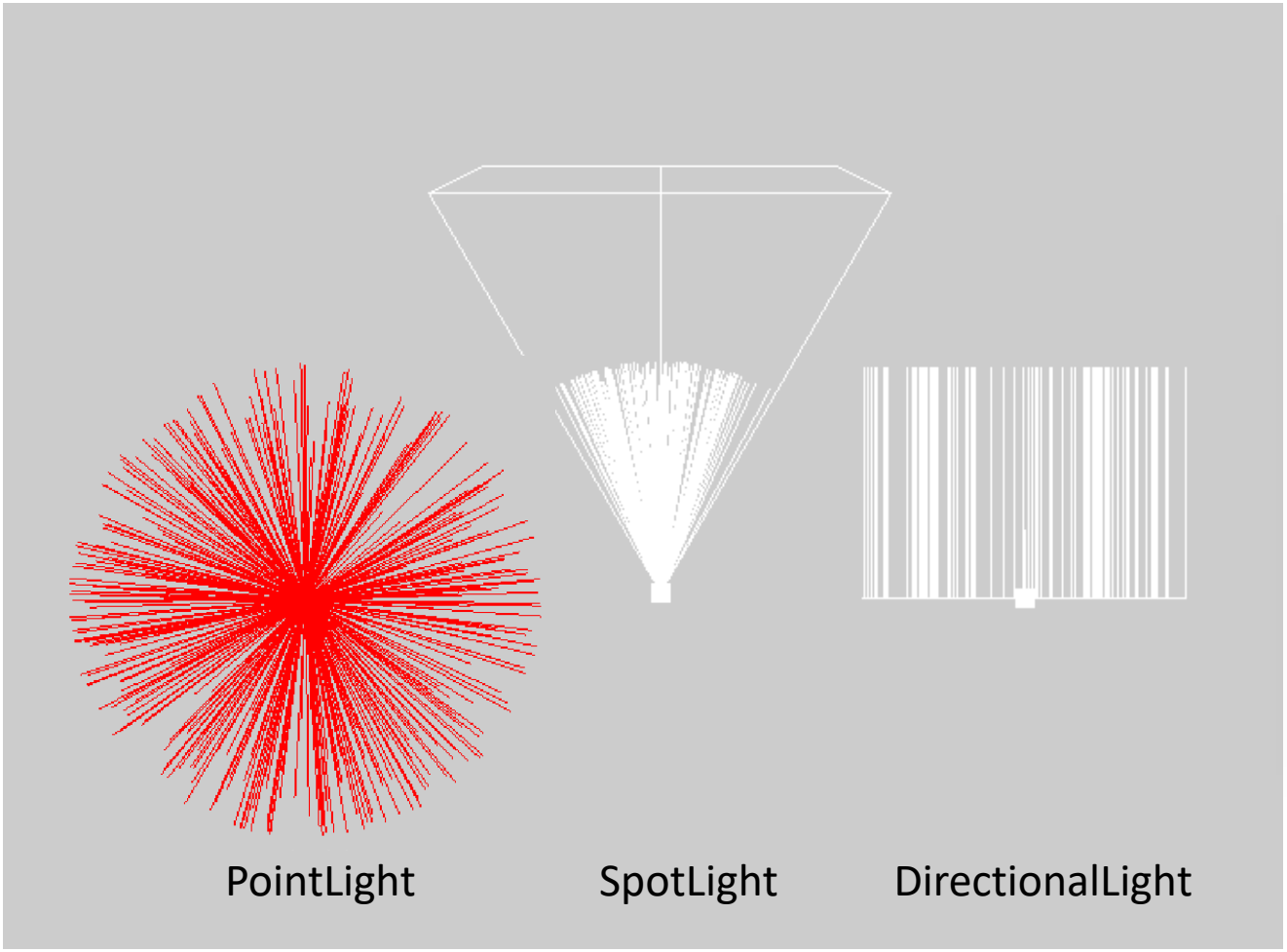
Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-introduction

Light modelling – Aspects: Light sources



Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-light-sources

Light modelling – Aspects: Light sources

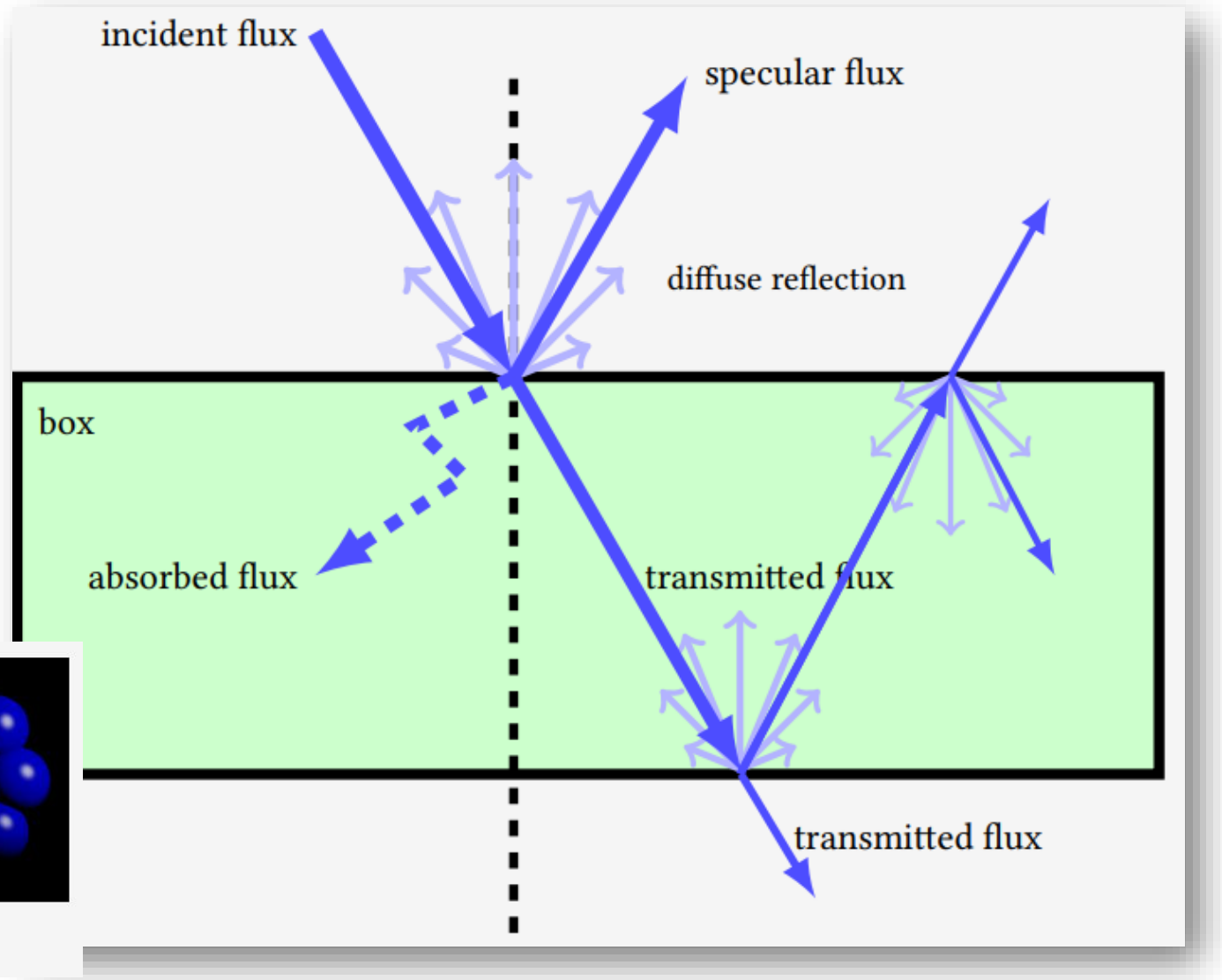



```
protected void init() [
  Axiom ==>
    LightNode.(setLight(new PointLight().(
      setPower(100),
      setVisualize(true),
      setNumberofrays(400),
      setRaylength(0.5),
      setColor(new RGBColor(1,0,0))
    )));
]
```

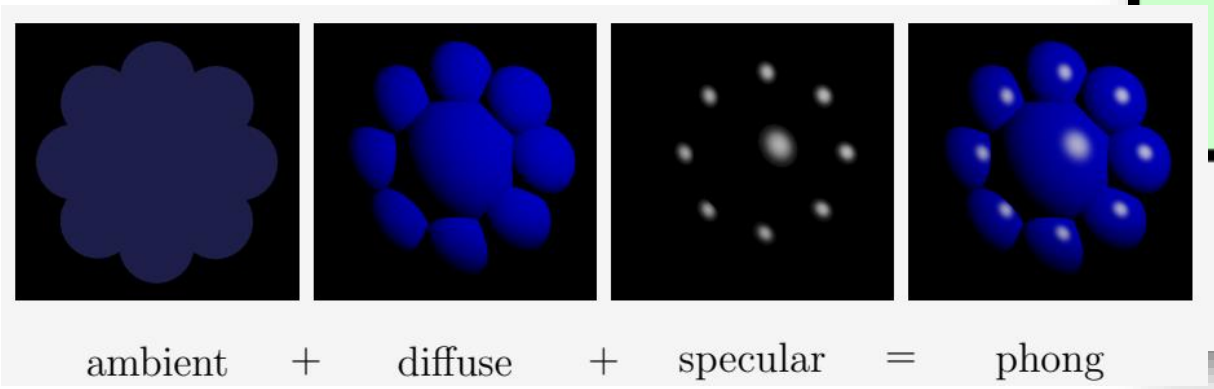
Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-light-sources

Light modelling – Aspects: Local illumination models

- Definition of optical properties of objects
- In computer graphics local illumination model are called shaders
- Shaders define the amount of absorption, reflection and transmission and how the light rays are scattered



Phong reflection model, B.T. Phong, 1973



Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-light-shader

Light modelling – Aspects: Local illumination model

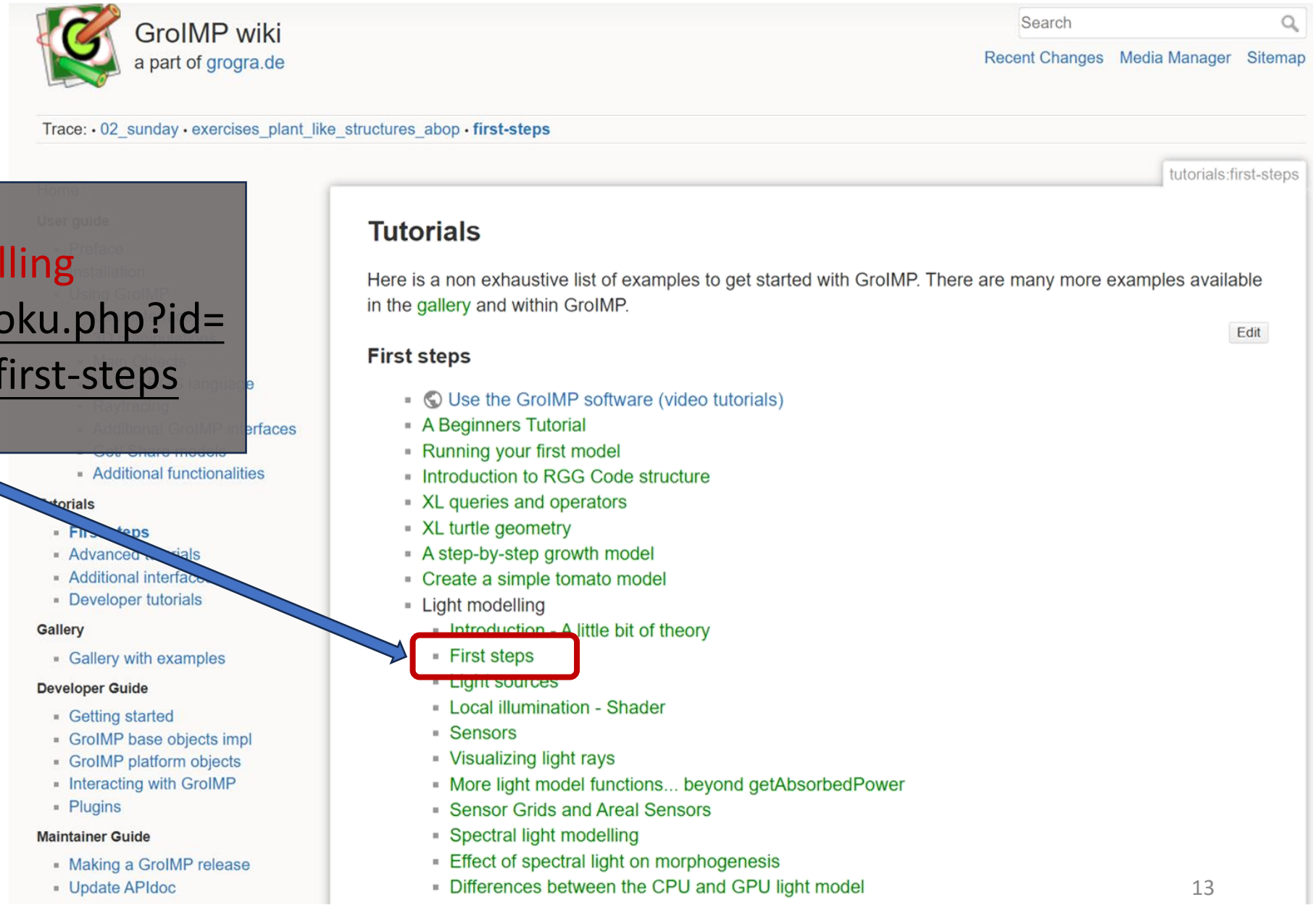
1. Define a Phong shader and set some properties
2. Apply the shader to a Box object
3. Note: absorbed = received – diffuse(reflected) – transmitted

```
static Phong myShader = new Phong();
static {
  myShader.setDiffuse(new RGBColor(0.1, 0.9, 0.0));
  myShader.setTransparency(new Graytone(0.1));
  myShader.setSpecular(new Graytone(0.1));
  myShader.setShininess(new Graytone(0.05));
}

protected void init () [
  Axiom ==> Box(0.001,1,1).(setShader(myShader));
]
```

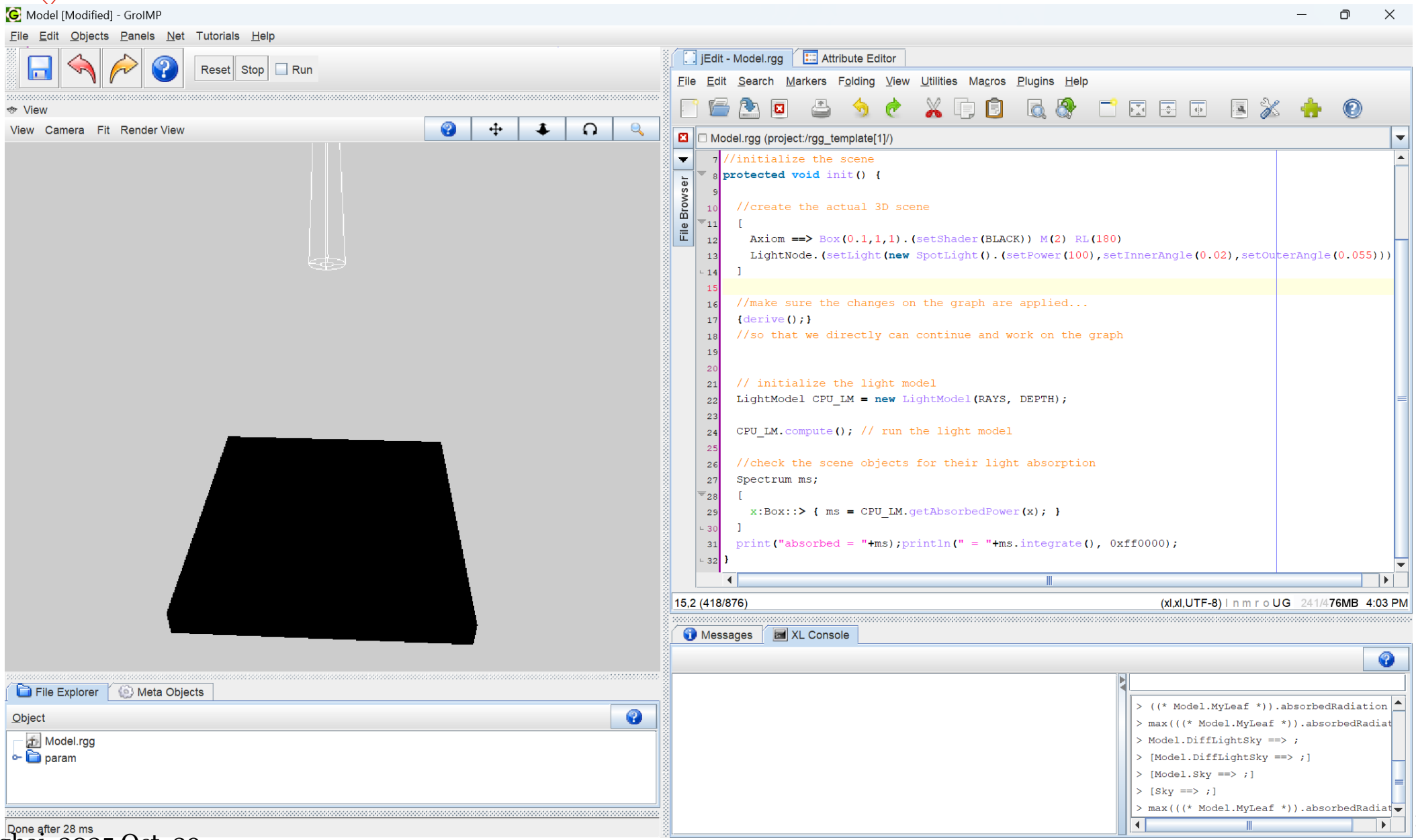
Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-light-shader

Light modelling – Hands-on

First steps on light modelling
https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-first-steps



Light modelling – Hands-on



Manipulate the model

- Change the colour of the box object to, e.g., red, and see how the absorption changes

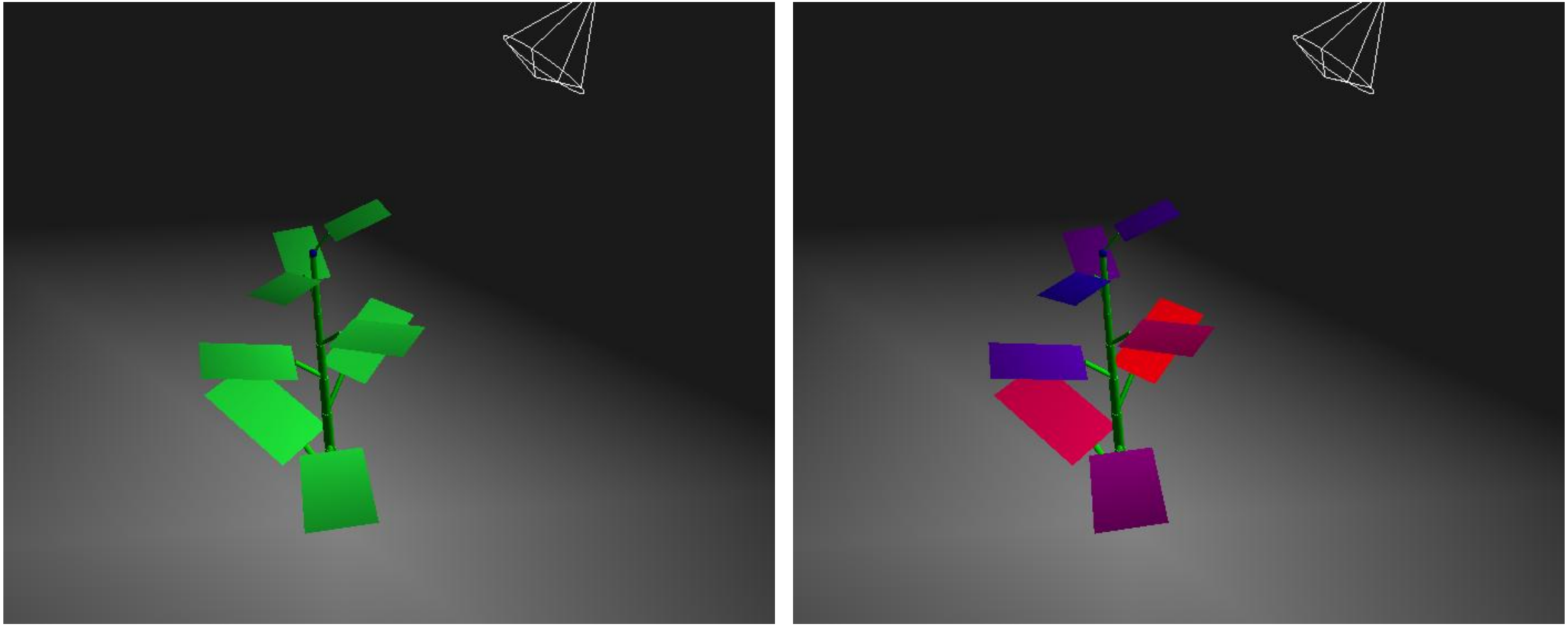
```
11  [
12    Axiom ==> Box(0.1,1,1).(setShader(RED)) M(2) RL(180)
13    LightNode.(setLight(new SpotLight().(setPower(100),setInnerAngle(0.02),
14  ]
```

- Change the colour of the box to some arbitrary colour and try to understand output.

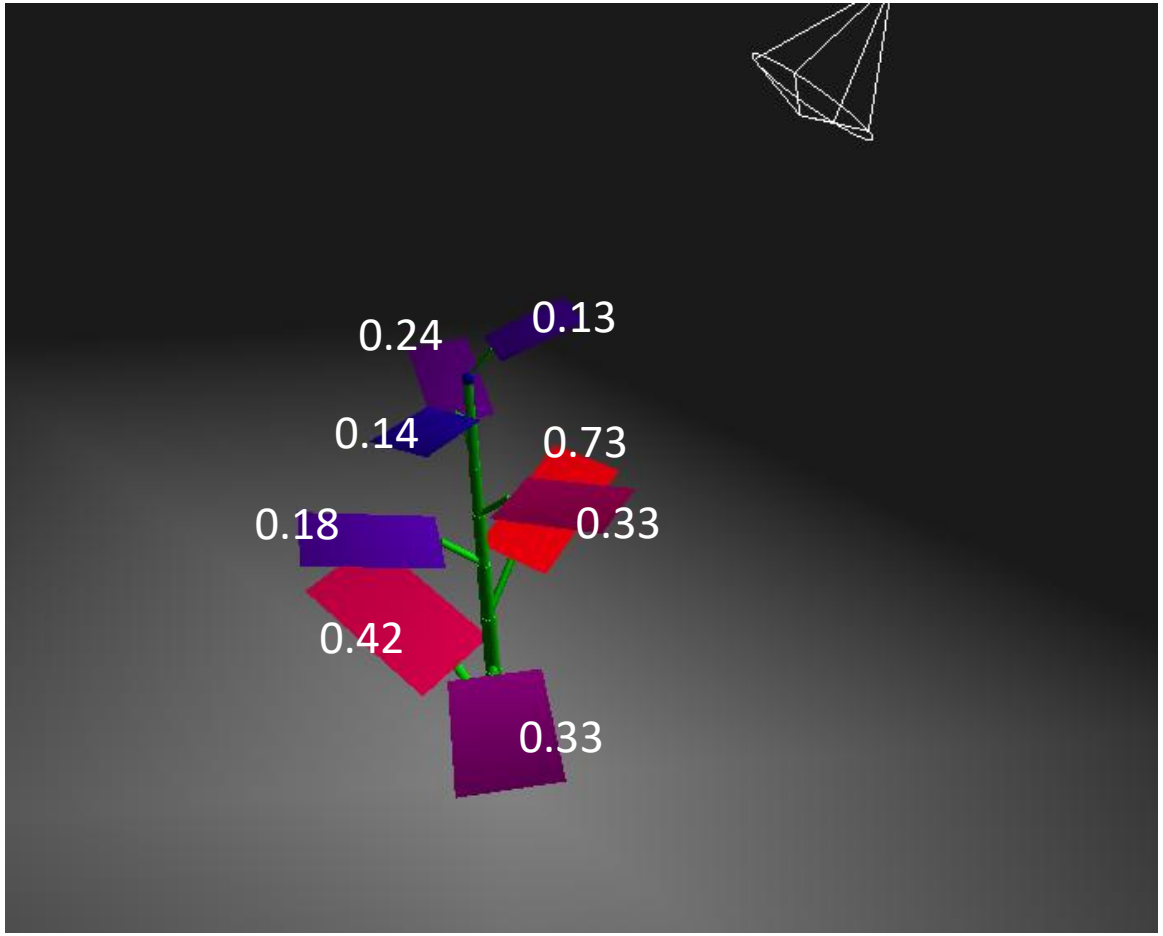
```
11  [
12    Axiom ==> Box(0.1,1,1).(setShader(new RGBAShader(0.1, 0.5, 0.3))) M(2)
13    LightNode.(setLight(new SpotLight().(setPower(100),setInnerAngle(0.02),
14  ]
```

```
absorbed = (29.99999995033074, 16.666666666665247, 23.33333293596735) = 69.99999955296333
```

Simulating light absorption at organ level



Using the output of the light modelling as input for photosynthesis estimation



- Individual absorption values [Watt]
- Converting to PAR/PPFD

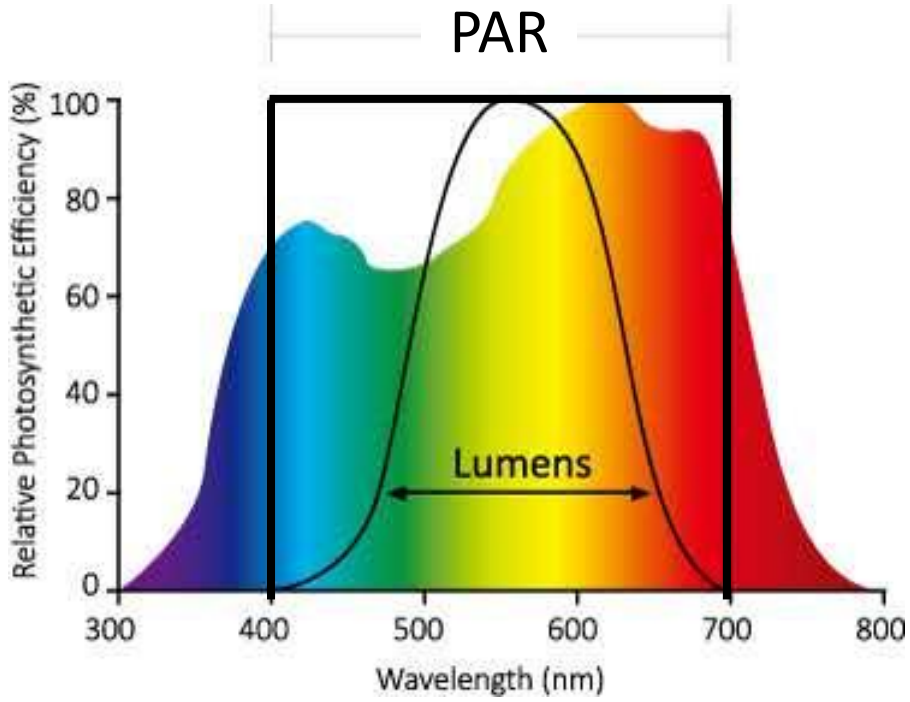
```
netPhotosynthesis = PSModel.calculatePhotosynthesis(
        airTemp,
        relativeHumidity,
        CO2_ppm,
        leafAge,
        PPFD  ← here comes the light absorption in
);
```

PSModel … Photosynthesis model, e.g., any FvCB implementation or simple light response curves, e.g., *KimLieth, Thornley, Baldocchi, Host, Marshall & Biscoe, LiehtPasian, JonsonThornley, Mueller & Echenroerder, etc.*

PAR vs PPDF

PAR, Photosynthetic Active Radiation, defines a range of wavelengths/light of the spectrum of (visible) light that plants can use for photo-synthesis. PAR is NOT a measurement or “metric” like centimeter, or kilo. Typically, PAR is defined as the range of 400 to 700nm.

PPFD, Photosynthetic Photon Flux Density, is the measurement of the amount of light (how many PAR photons), actually hit a specific surface area every second. PPFD is expressed in µmol/m2/s

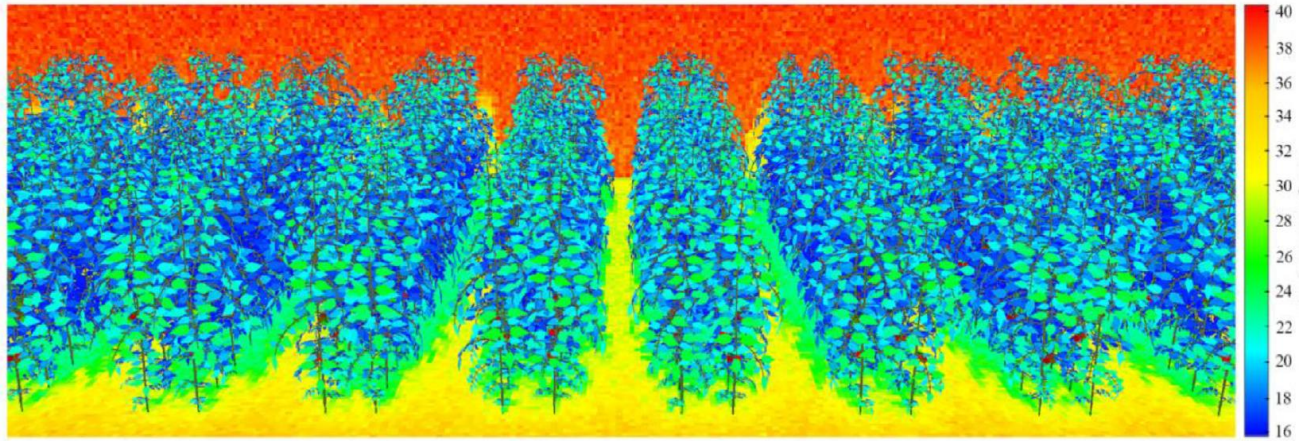


PAR is the “what kind of light”
PPFD is the “ how much of it”

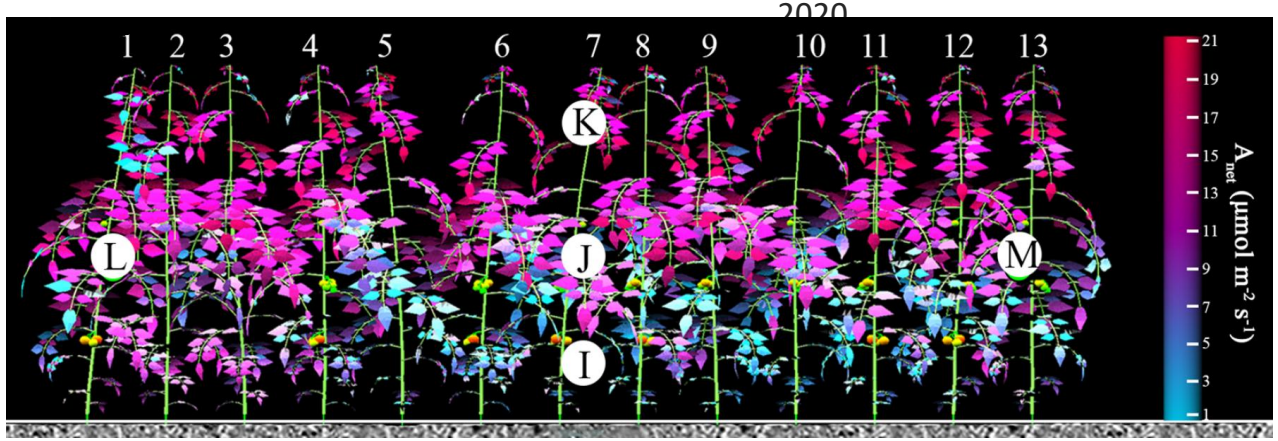
Light modelling – Examples

崖州湾国家实验室
Yazhouwan National Laboratory

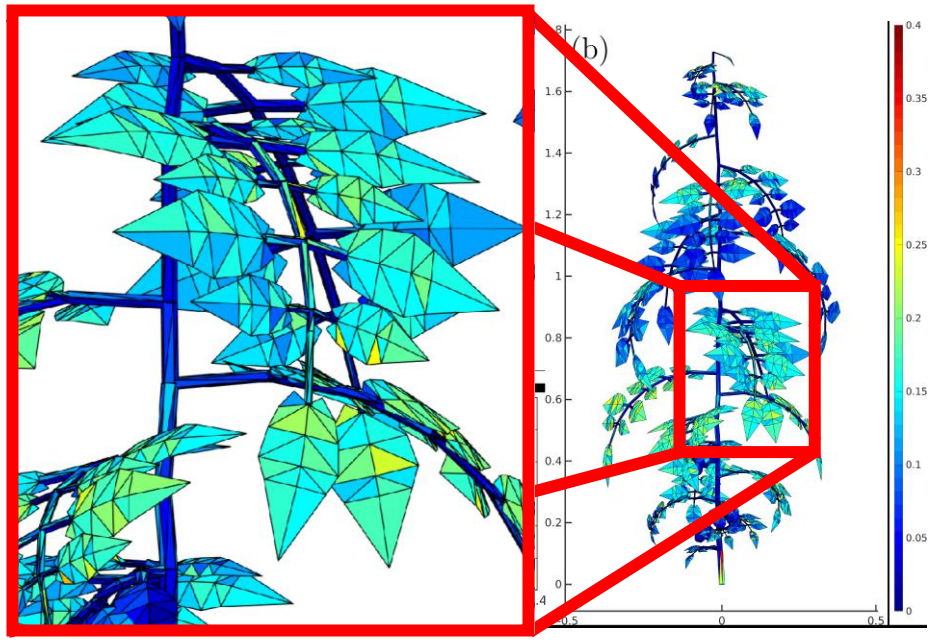
Detailed calculation of light microclimate at organ level (… and even below if needed)



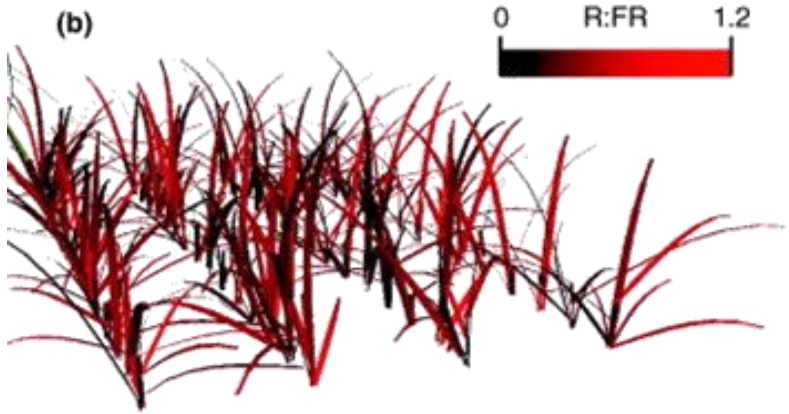
Zhang et al., Renewable Energy, 2020



Zhang et al., Agricultural and Forest Meteorology, 2021



Henke & Gladilin, Remote Sens., 2022

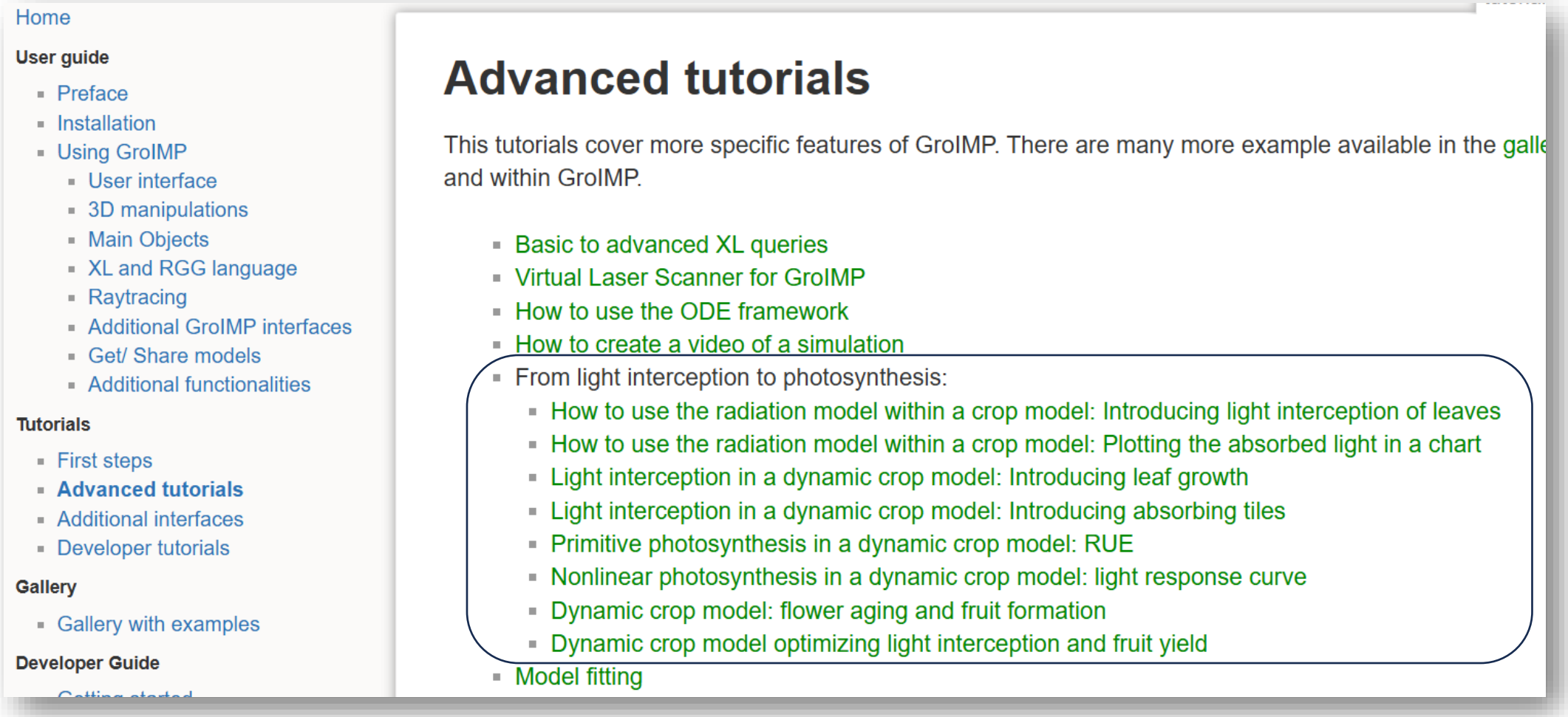


Evers et al., Trends in Plant Science, 2011

Light modelling – Linking to photosynthesis

崖州湾国家实验室 Yazhouwan National Laboratory

Series of tutorials on ‘Light interception to PS’ in the **Advanced tutorials** on the wiki pages



Home

User guide

- Preface
- Installation
- Using GroIMP
  - User interface
  - 3D manipulations
  - Main Objects
  - XL and RGG language
  - Raytracing
  - Additional GroIMP interfaces
  - Get/ Share models
  - Additional functionalities

Tutorials

- First steps
- **Advanced tutorials**
- Additional interfaces
- Developer tutorials

Gallery

- Gallery with examples

Developer Guide

- Getting started

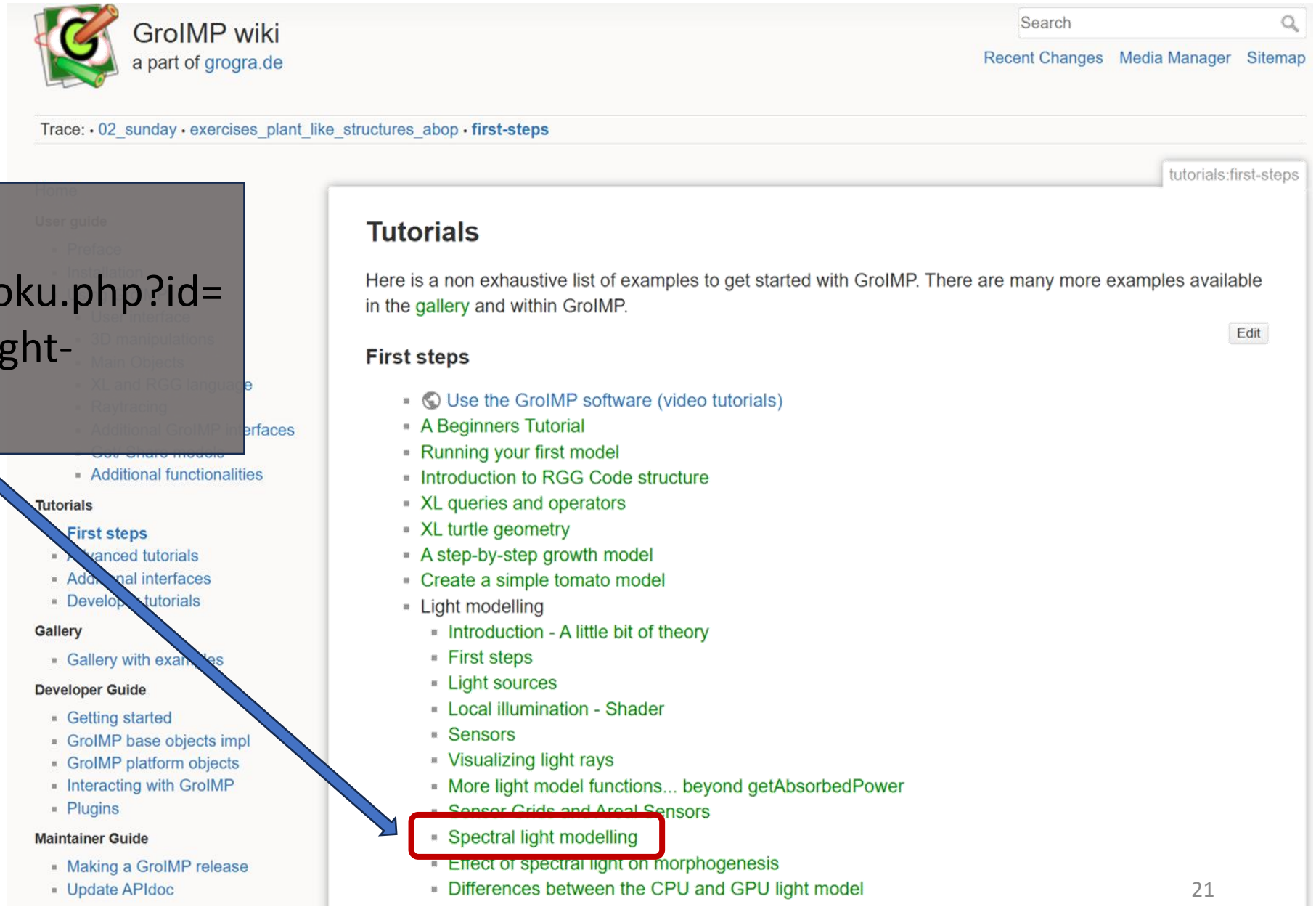
Advanced tutorials

This tutorials cover more specific features of GroIMP. There are many more example available in the galle and within GroIMP.

- Basic to advanced XL queries
- Virtual Laser Scanner for GroIMP
- How to use the ODE framework
- How to create a video of a simulation
- From light interception to photosynthesis:
  - How to use the radiation model within a crop model: Introducing light interception of leaves
  - How to use the radiation model within a crop model: Plotting the absorbed light in a chart
  - Light interception in a dynamic crop model: Introducing leaf growth
  - Light interception in a dynamic crop model: Introducing absorbing tiles
  - Primitive photosynthesis in a dynamic crop model: RUE
  - Nonlinear photosynthesis in a dynamic crop model: light response curve
  - Dynamic crop model: flower aging and fruit formation
  - Dynamic crop model optimizing light interception and fruit yield
- Model fitting

Full spectral light modelling

Spectral light modelling
https://wiki.grogra.de/doku.php?id=tutorials:basic-spectral-light-modeling

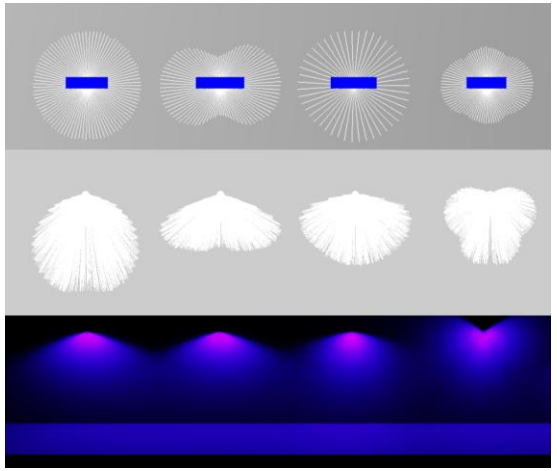


Light Simulations – LED Modelling 光照模型

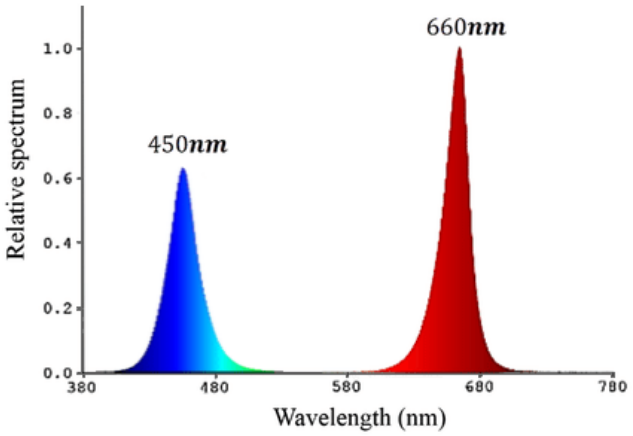
Full spectral Light simulations
全光谱光线追踪

崖州湾国家实验室 Yazhouwan National Laboratory

Single LED model（光珠）

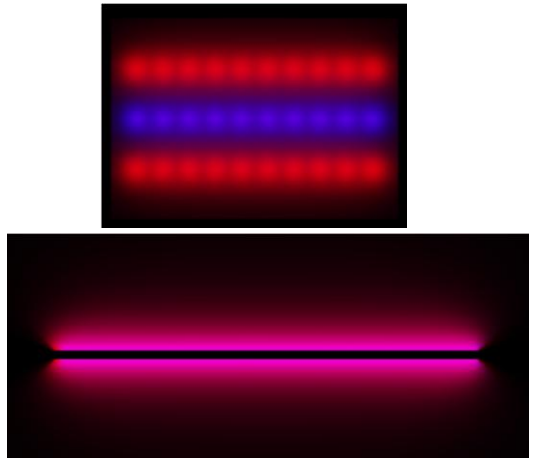


Physical Light Distribution
物理光分布



Spectral Power Distribution
光谱功率分布

Simulated double-string LED stripes（灯条）

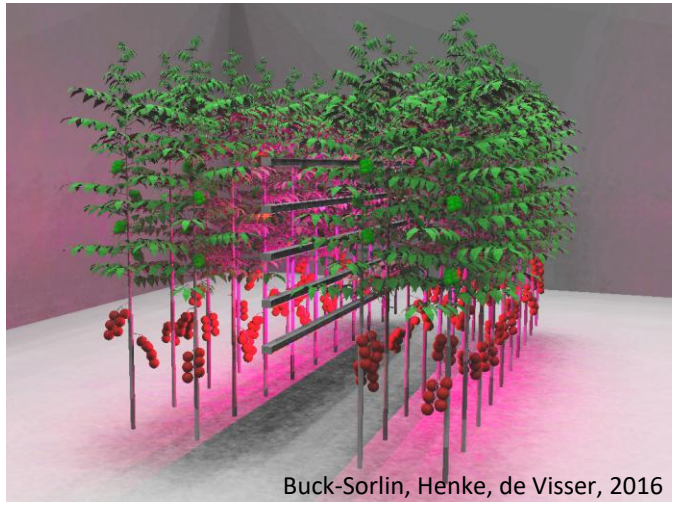


Simulated double-striped LED panel

Characteristics of Light
- Quality
- Quantity
- Duration

真实光照系统

LED模型



Buck-Sorlin, Henke, de Visser, 2016

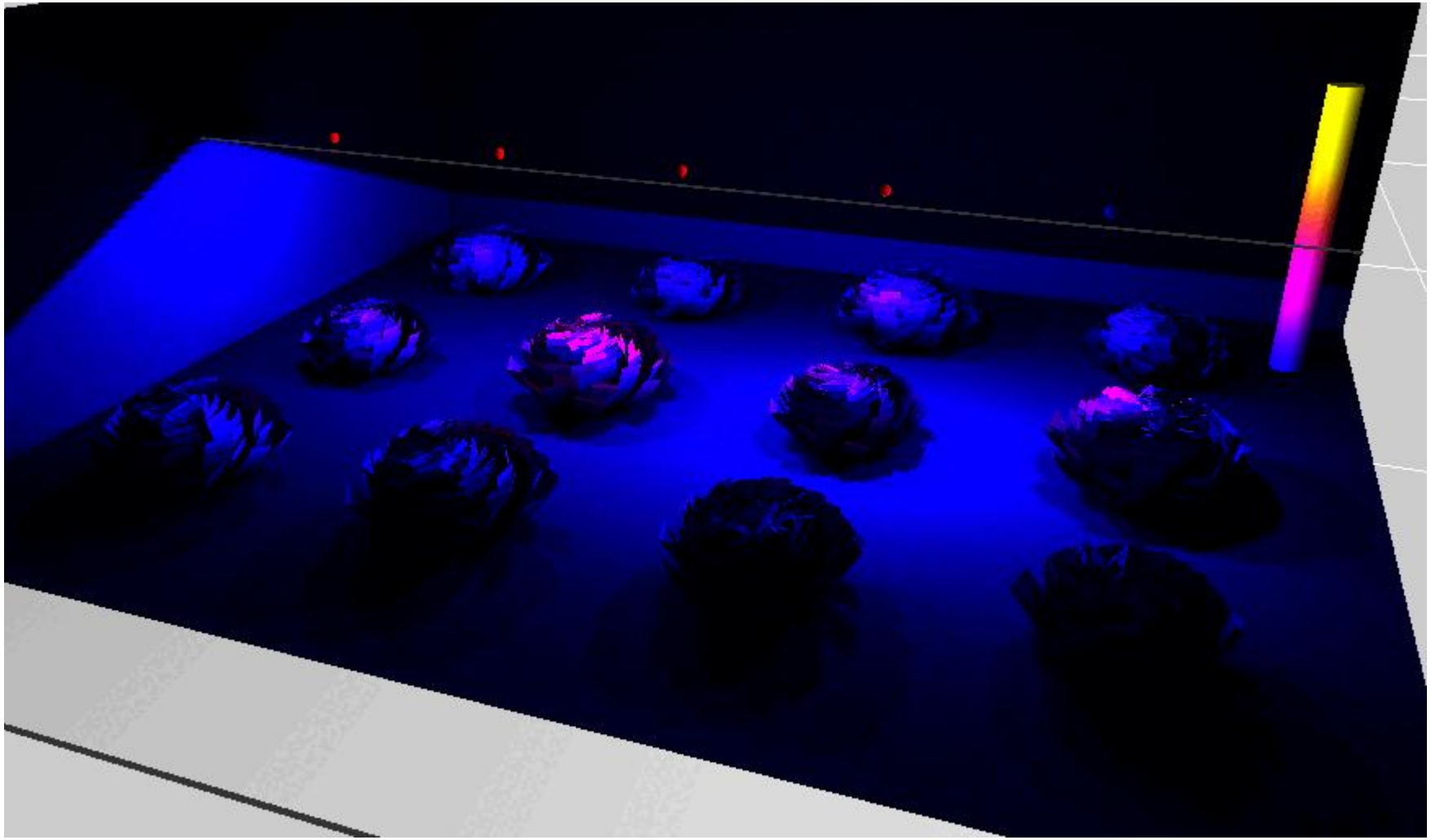
Light Simulations – LED Modelling 光照模型

Simulations of

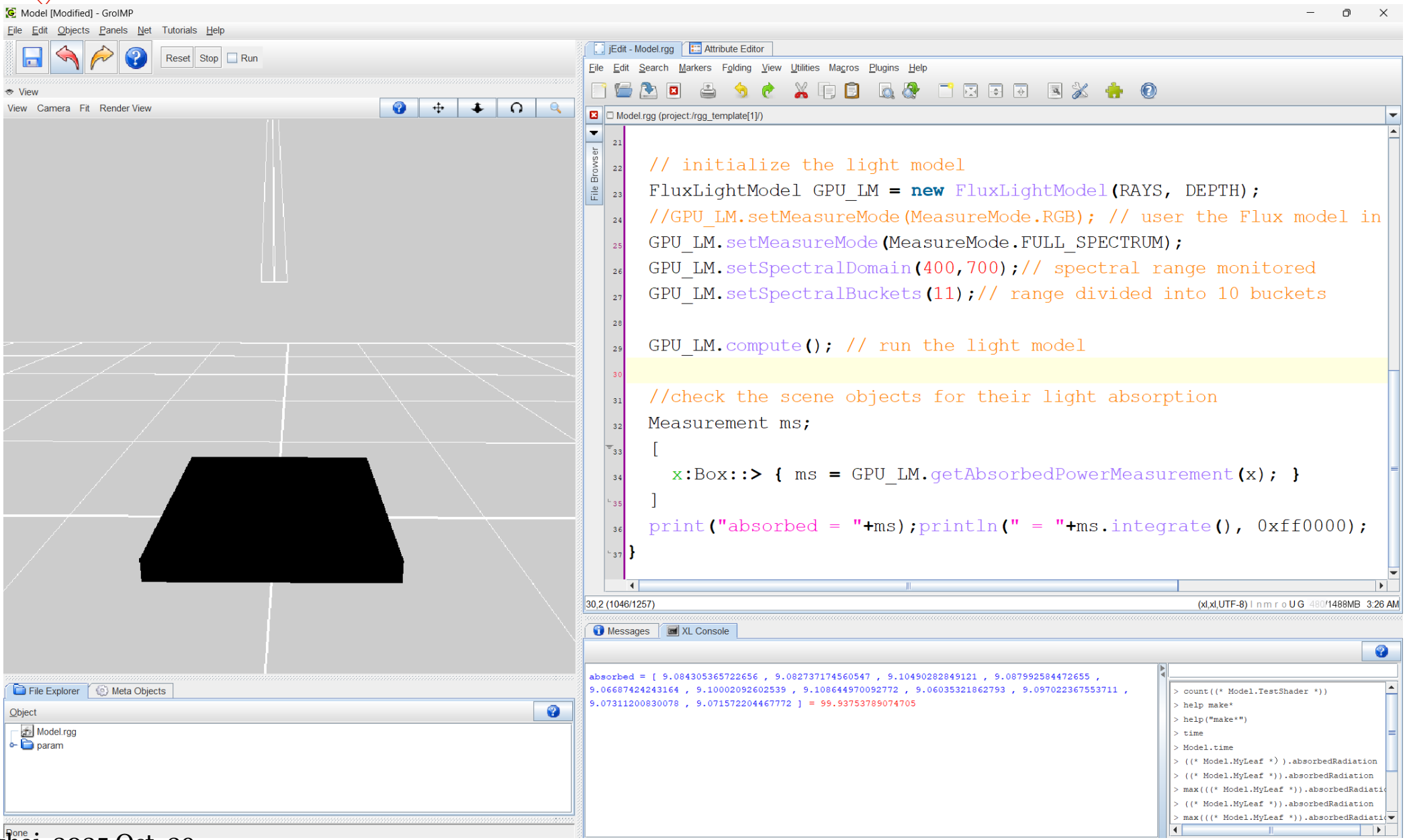
- Numbers of LEDs
  灯珠数量
- Numbers of LED stripes
  灯条数量
- Different PLDs
  物理光分布
- Different SPDs
  光谱功率分布
- Different arrangements

Obtain

- Detailed information about light-micro climate
  详细的光照微环境



Light modelling – Hands-on – spectral!



Sensor nodes

“Invisible” nodes to sense radiation without interfering with the scene

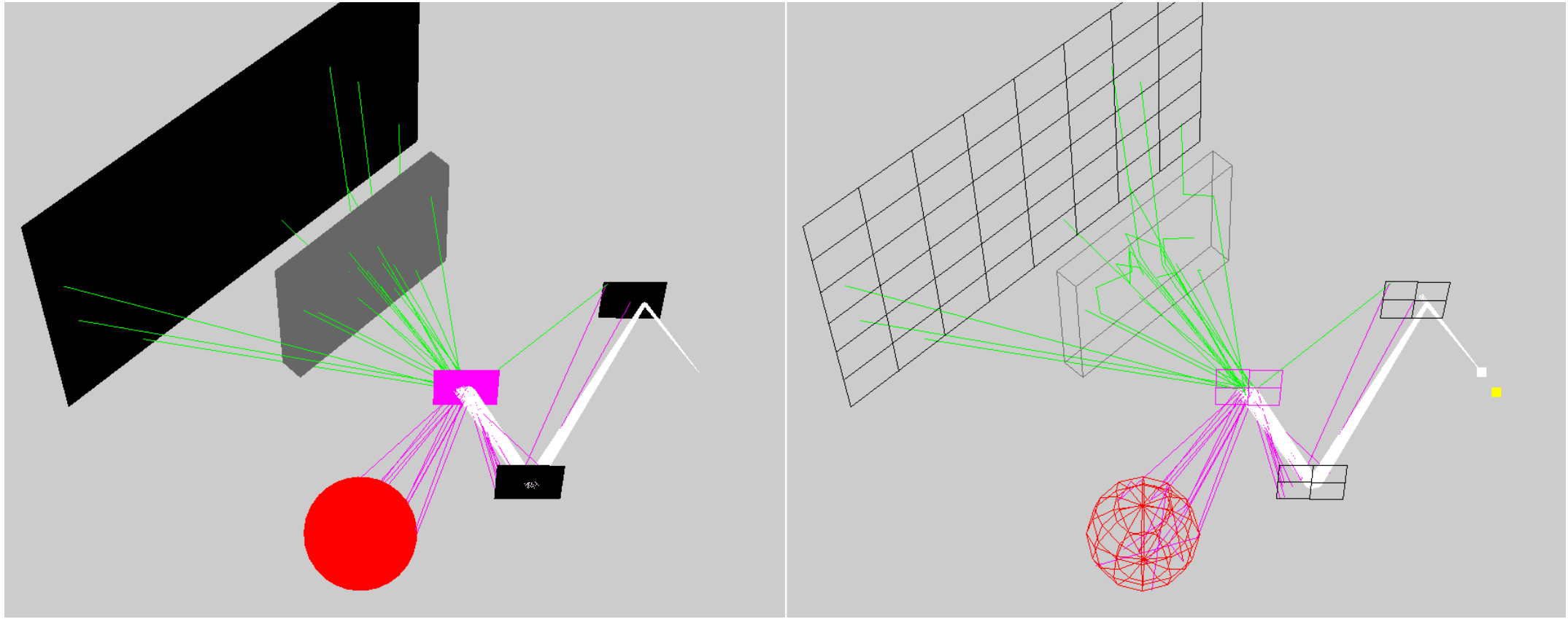
- Retuning absorption per square meter

```
import de.grogra.gpuflux.scene.experiment.Measurement;

// create a 5cm, white sensor node
Axiom ==> SensorNode().(setRadius(0.05), setColor(1, 1, 1));

const FluxLightModel GPU_LM = new FluxLightModel(RAYS, DEPTH);
// enable dispersion
GPU_LM.setDispersion(true);

//check what the sensor node has sensed
x:SensorNode ::> {
    Measurement spectrum = GPU_LM.getSensedIrradianceMeasurement(x);
    float absorbedPower = spectrum.integrate();
    ...
}
```

Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-sensors

Visualizing light rays


```
protected void init () [
    Axiom ==> LightModelVisualizer(175, 10);
]
```

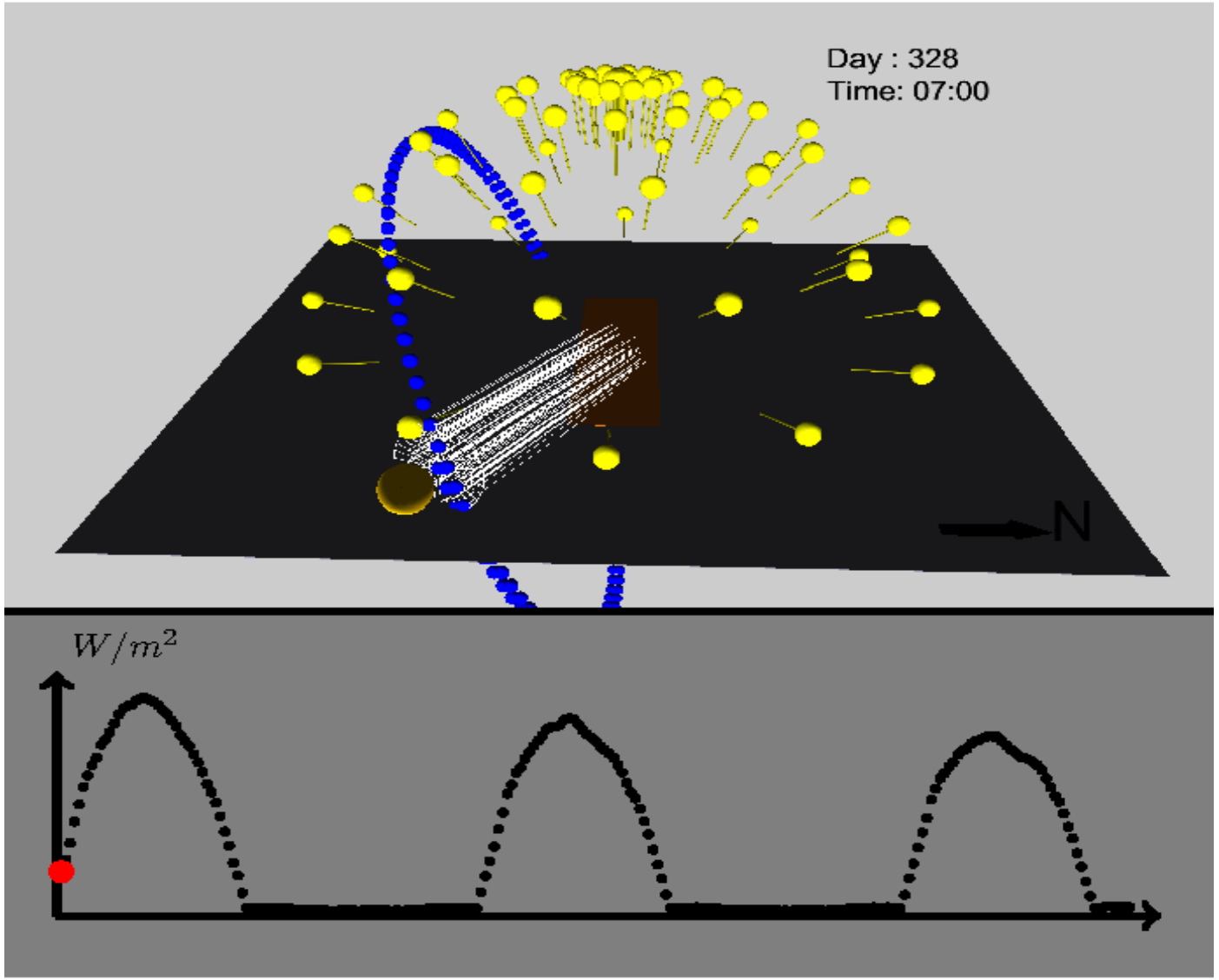


Wiki: https://wiki.grogra.de/doku.php?id=tutorials:light-modeling-visualizing-light-rays

Light dome

Direct and diffuse light dome

- Direct sun light
  (one direct light node)
- Diffuse sky light
  (hemisphere with 72 direct light nodes)



感谢观看
Thank you!

GroIMP WeChat Group

Valid until 11/5