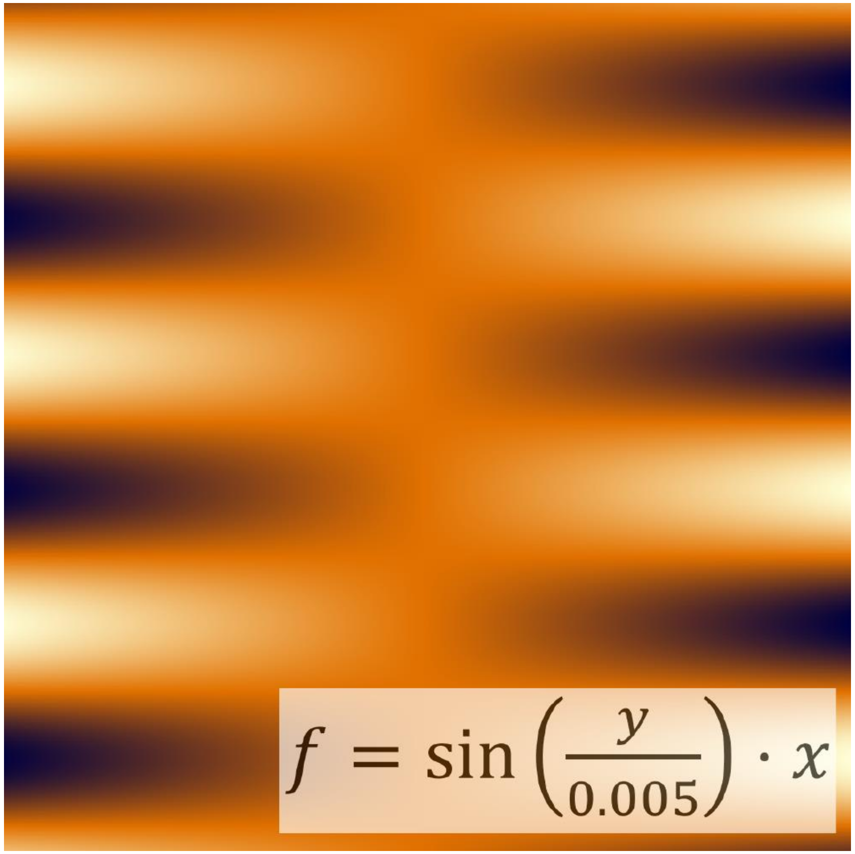
Programmable Light Source, Function, Interface and Medium

Abstract

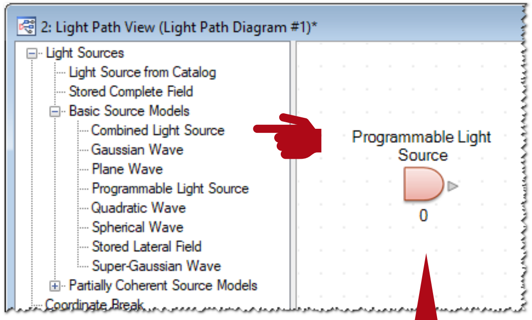


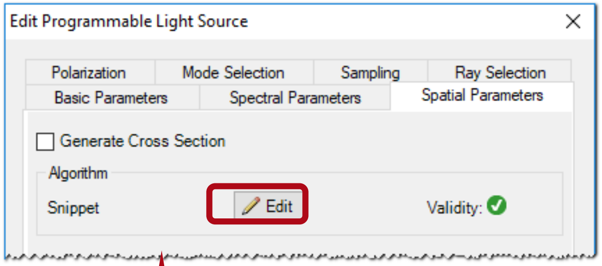
VirtualLab offers always different ways for the specification of optical objects, e.g., light source, interface, and medium. We provide predefined objects, the objects can be specified by measurement data or the object can be described by using programmable objects. The programming language for these customized objects is C#. VirtualLab supports the development of the programmable items by the source code editor. Different object have different input and return parameters. This use case explains the most specific parameters of the programmable light source, interface, function and medium.

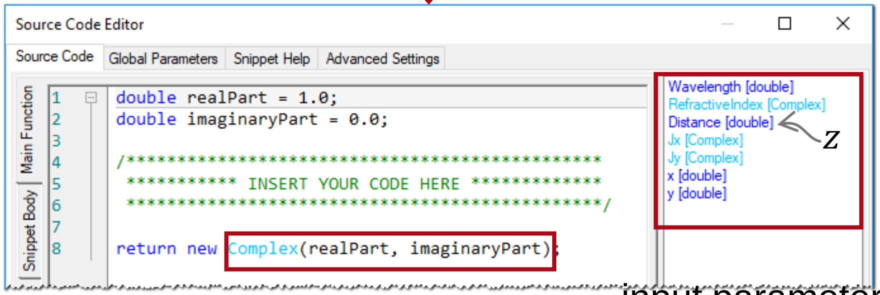
Programmable Light Source

- *Programmable Light Source* allows users to define the spatial distribution of a global polarized light mode in one plane. The mathematical representation is $\breve{E}_0(x, y, z, \breve{\boldsymbol{J}}, \lambda, \breve{n}(\lambda))$
  - (x, y, z) is the spatial coordinators
  - $\breve{\boldsymbol{J}} = \begin{pmatrix} \breve{J}_x \\ \breve{J}_y \end{pmatrix}$ represents the Jones vector

    $$\boldsymbol{E} = \begin{pmatrix} \breve{E}_x \\ \breve{E}_y \end{pmatrix} = \breve{E}_0 \cdot \begin{pmatrix} \breve{J}_x \\ \breve{J}_y \end{pmatrix}$$
  - λ is wavelength
  - $\breve{n}(\lambda)$ is the refractive index of the surrounding medium







$\breve{E}_0$

input parameters: $x, y, z, \boldsymbol{J}, \lambda, \breve{n}(\lambda)$

Example of Programmable Light Source

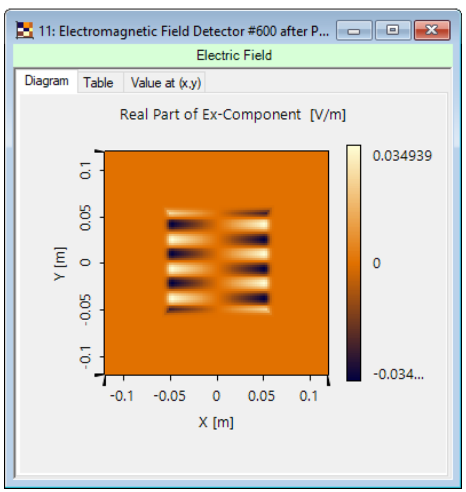
C# snippet

```
double realPart = 1.0;
double imaginaryPart = 0.0;

realPart = Math.Sin(y / 5e-3) * x;

return new Complex(realPart, imaginaryPart);
```

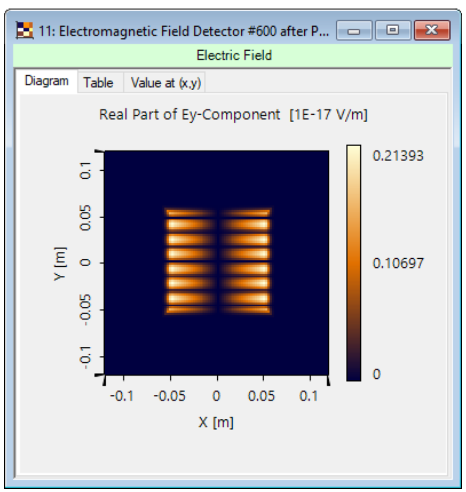
Source parameters

parameter	value
size	100 mm x 100 mm
polarization	right circularly polarized

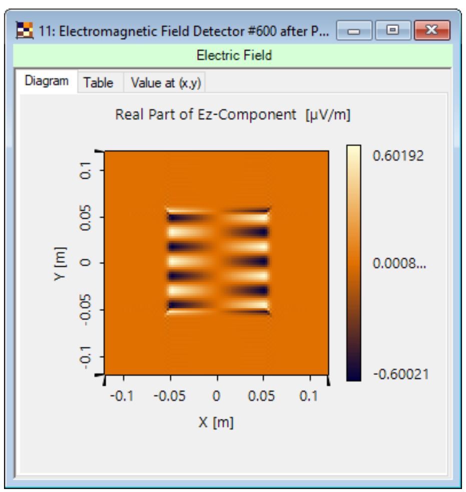
electric field [V/m]



$\Re(E_x)$



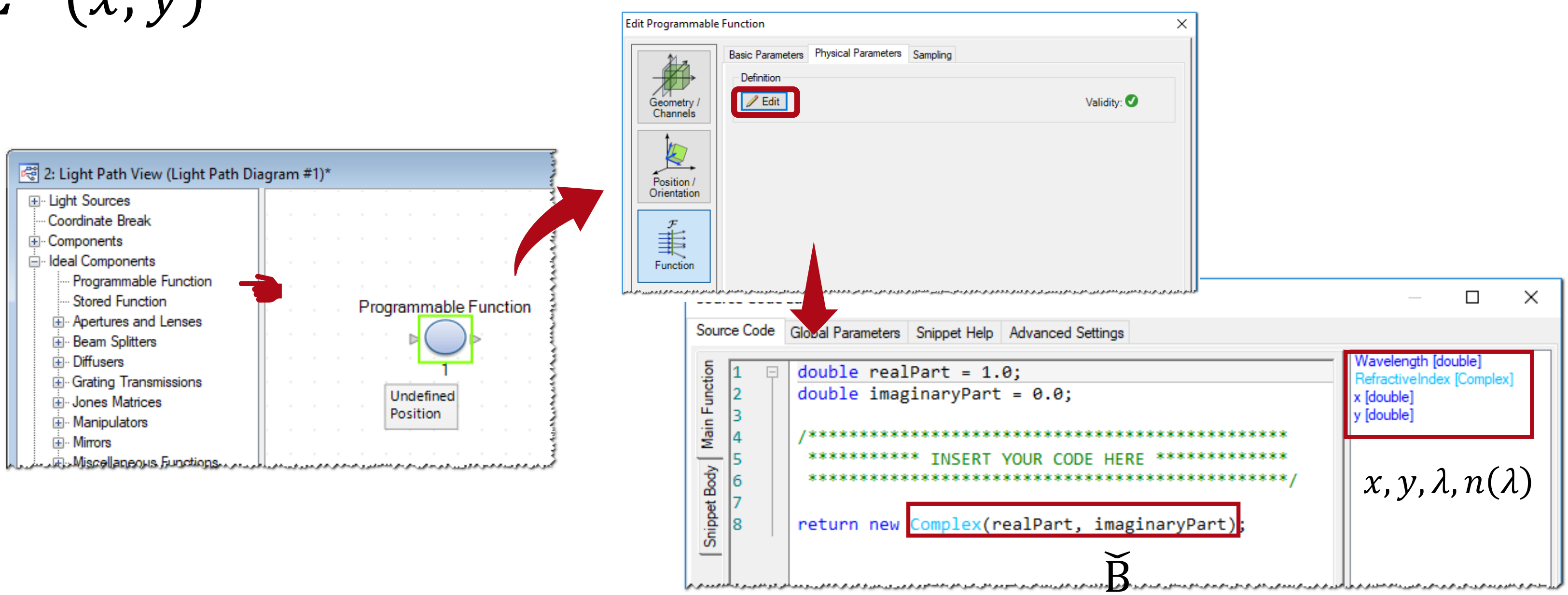
$\Re(E_y)$



$\Re(E_z)$

Programmable Function

- Programmable function allows users to define a specific boundary response, which is mathematically represented as $\breve{\mathrm{B}} = f(x, y, \lambda, \breve{n}(\lambda))$.
- Output field of *Programmable Function* is: $\boldsymbol{E}^{\mathrm{out}}(x, y) = \breve{\mathrm{B}} \cdot \boldsymbol{E}^{\mathrm{in}}(x, y)$

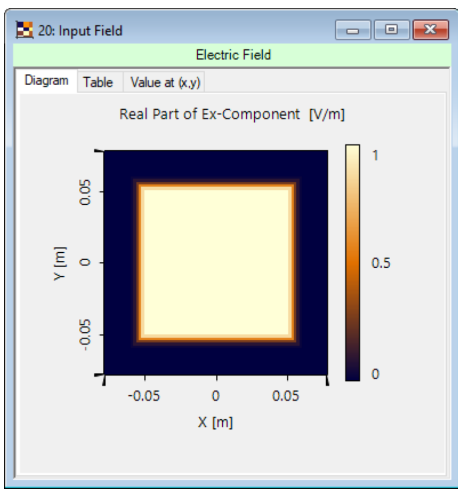


Example of Programmable Function

C# snippet

```
double realPart = 1.0;
double imaginaryPart = 0.0;

realPart = Math.Sin(y / 5e-3) * x;

return new Complex(realPart, imaginaryPart);
```

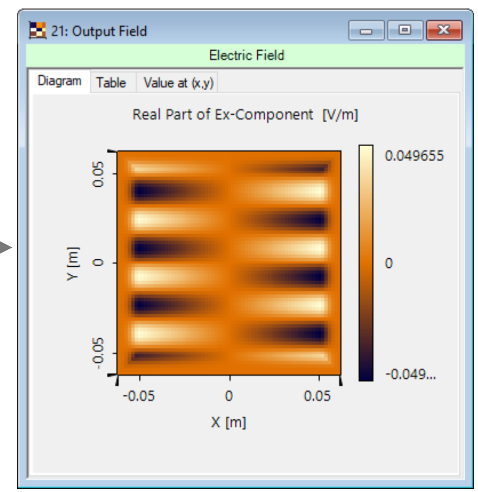
input field



$\Re(E_x^{\mathrm{in}})$

programmable function

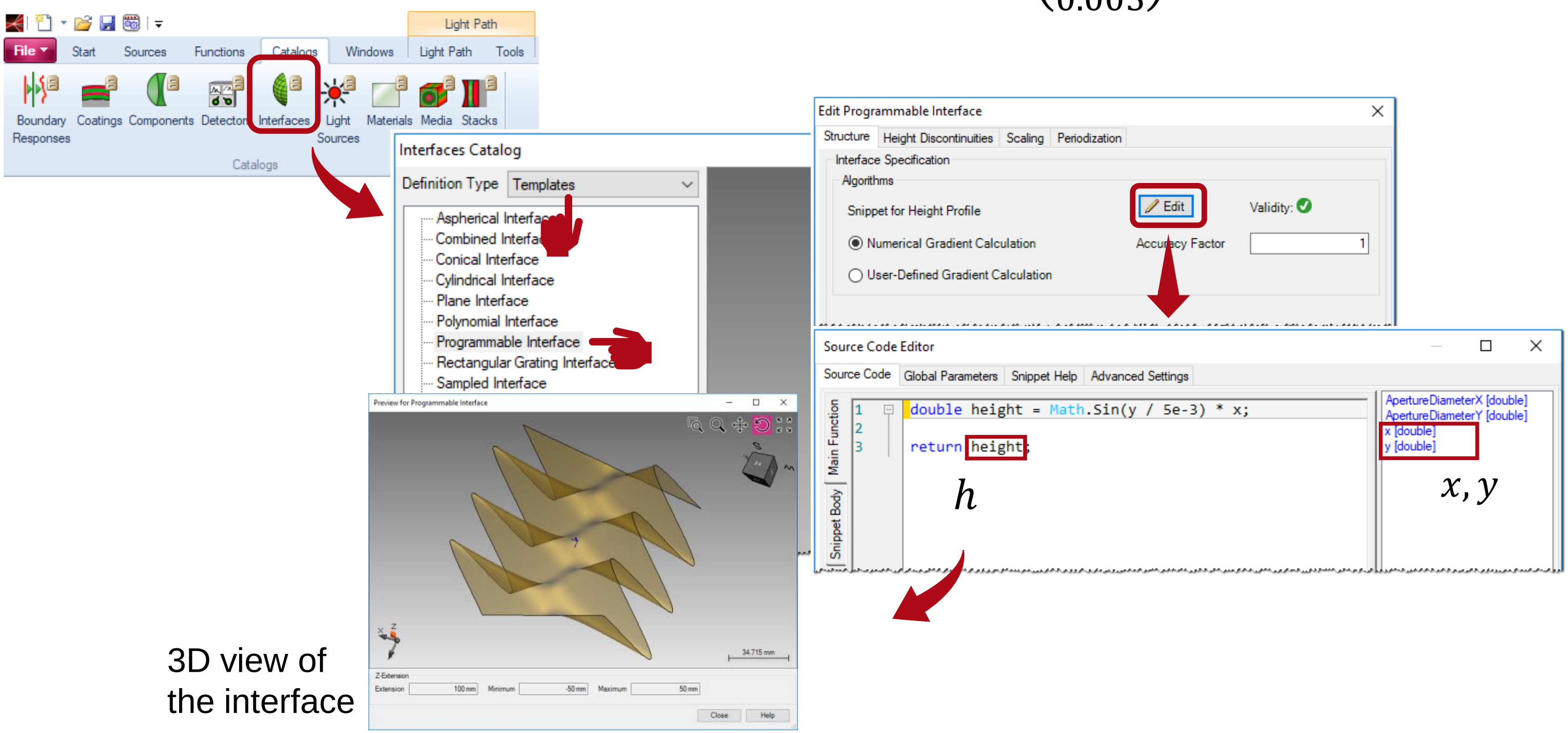
output field



$\Re(E_x^{\mathrm{out}})$

Programmable Interface

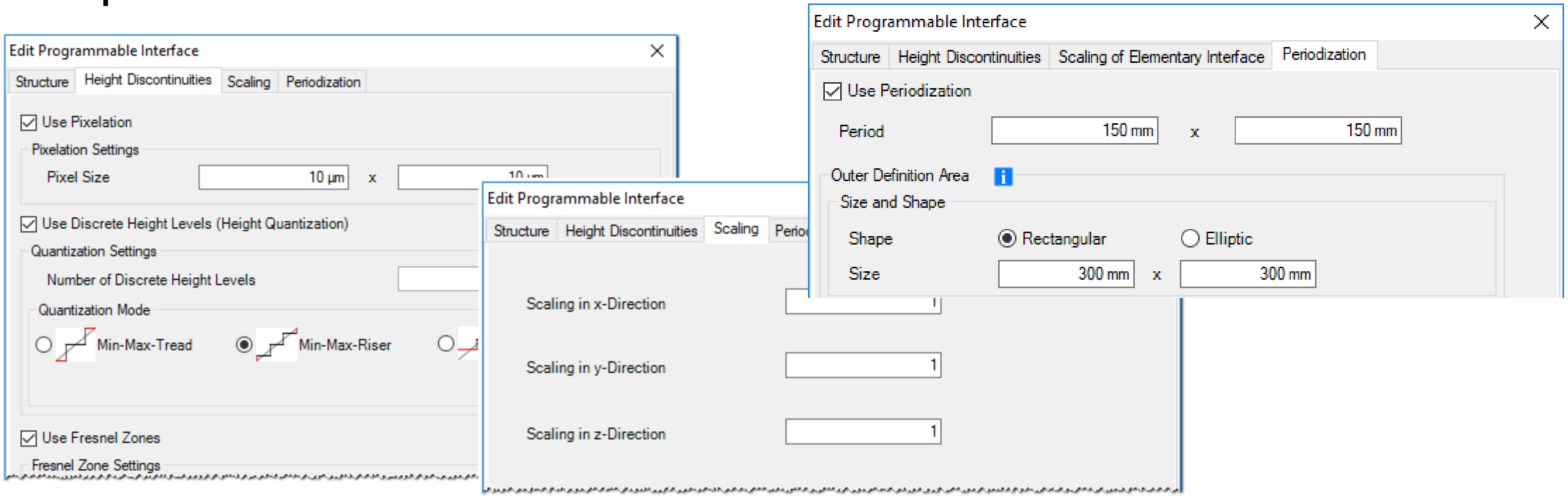
- *Programmable Interface* allows users to define a height profile of an interface.
  - Height profile $h(x,y)$, e.g., $h(x,y) = \sin\left(\frac{y}{0.005}\right) \cdot x$



3D view of the interface

Programmable Interface: Properties

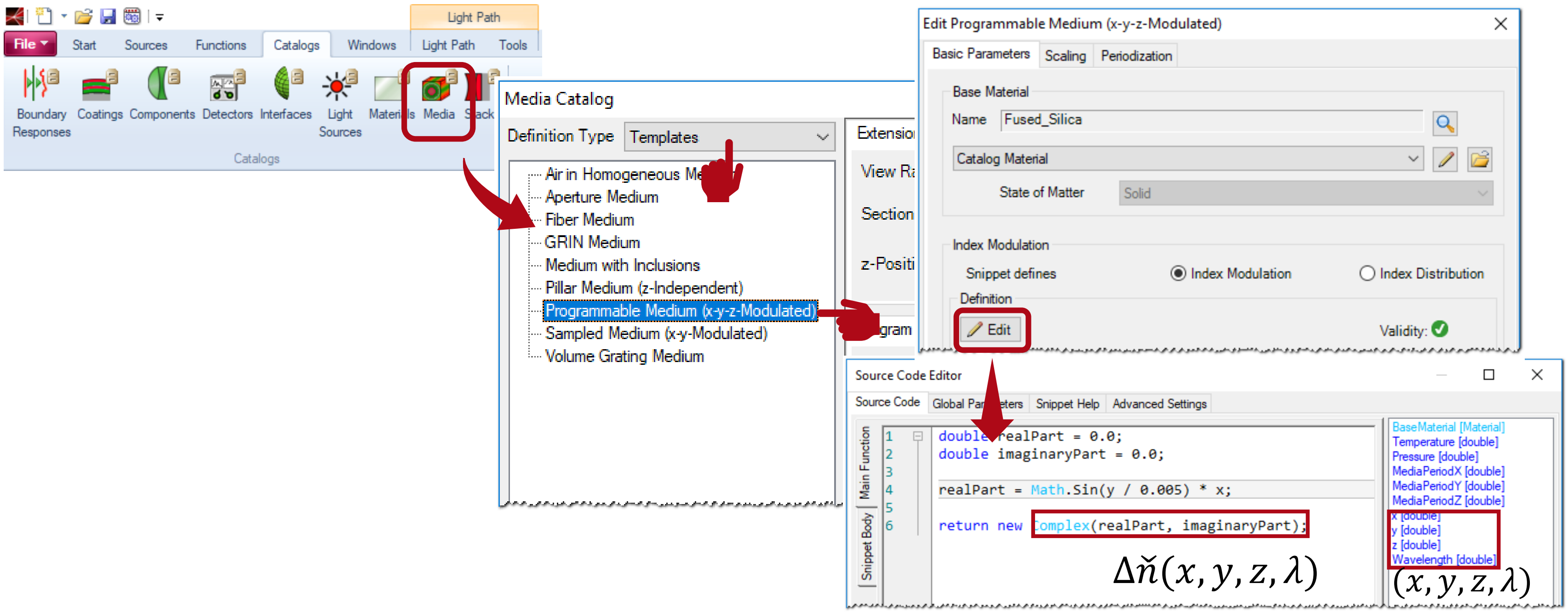
- *Programmable Interface* contains all properties of general interfaces in VirtualLab Fusion.
  - discontinuities: the interface can be quantized
  - scaling: stretch the interface in specific direction
  - periodization: make a repetition of the interface to generate a periodic interface.



More details can be found in “Getting started” tutorial, i.e., Catalogs II: Interfaces

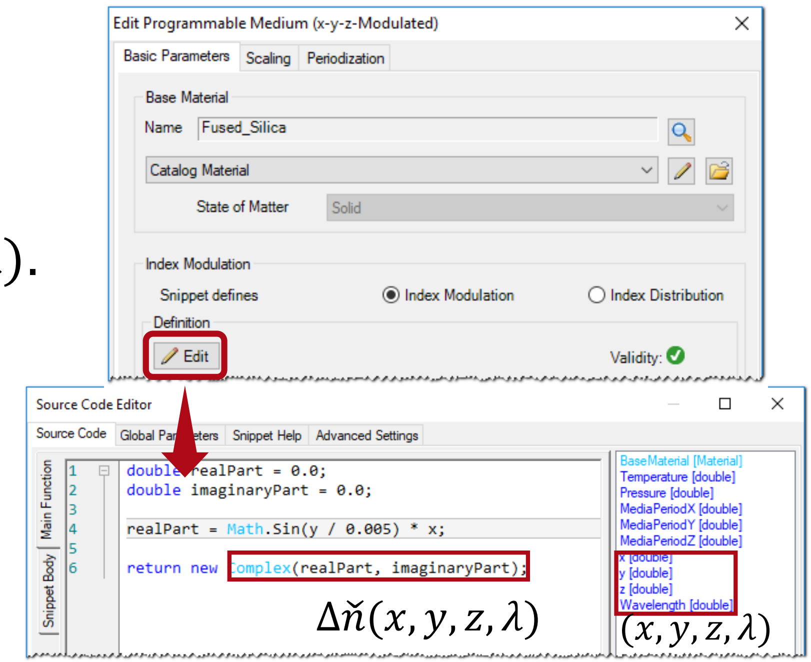
Programmable Medium

- *Programmable Medium* allows users to define an arbitrary, complex-valued refractive index distribution $\breve{n}(x, y, z, \lambda)$. Variance of refractive index $\Delta\breve{n}(x, y, z, \lambda)$ is programmed.



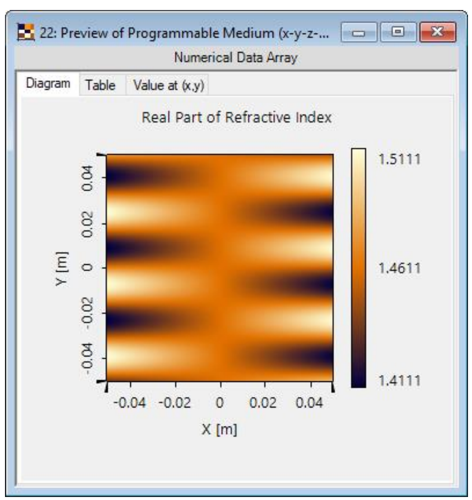
Programmable Medium

- *Programmable Medium* allows users to define an arbitrary, complex-valued refractive index distribution $\breve{n}(x, y, z, \lambda)$. $\Delta\breve{n}(x, y, z, \lambda)$ is programmed.
- There are two ways to define $\breve{n}(x, y, z, \lambda)$:
  - *Index Distribution*:
    $\breve{n}(x, y, z, \lambda) = \Delta\breve{n}(x, y, z, \lambda)$
  - *Index Modulation*:
    $\breve{n}(x, y, z, \lambda) = n_0(\lambda) + \Delta\breve{n}(x, y, z, \lambda)$.
    $n_0(\lambda)$ is given by *Base Material*.

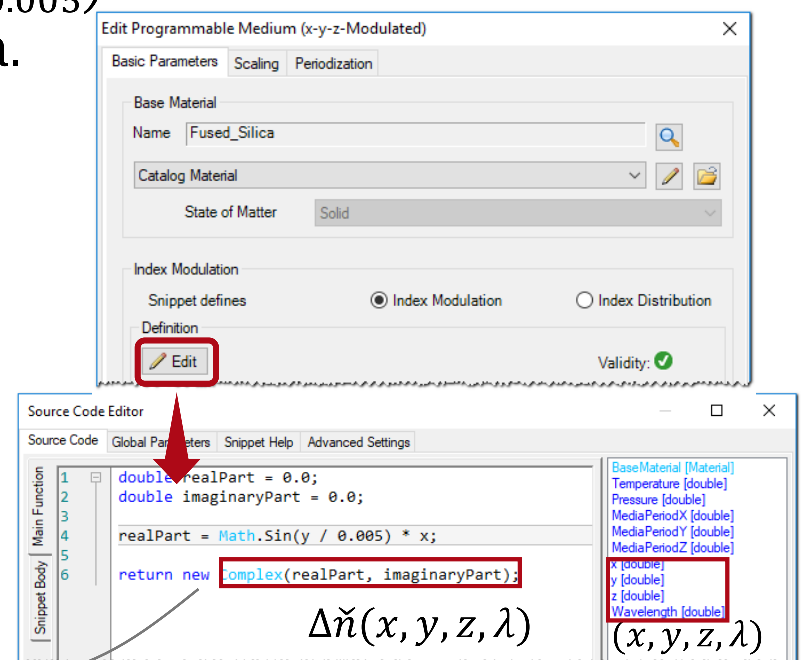


Example of Programmable Medium

- *Programmable Medium* allows users to define an arbitrary, complex-valued refractive index distribution $\breve{n}(x,y,z,\lambda)$.
  - E.g., $\breve{n}(x,y,z,\lambda) = \breve{n}^{\mathrm{FS}}(\lambda) + \sin\left(\frac{y}{0.005}\right) \cdot x$, with $\breve{n}^{\mathrm{FS}}(\lambda)$ denoting the refractive index of fused silica.



refractive index



Document Information

title	Programmable Light Source, Function, Interface and Medium
version	1.0
VL version used for simulations	7.0.3.4
category	Feature Use Case