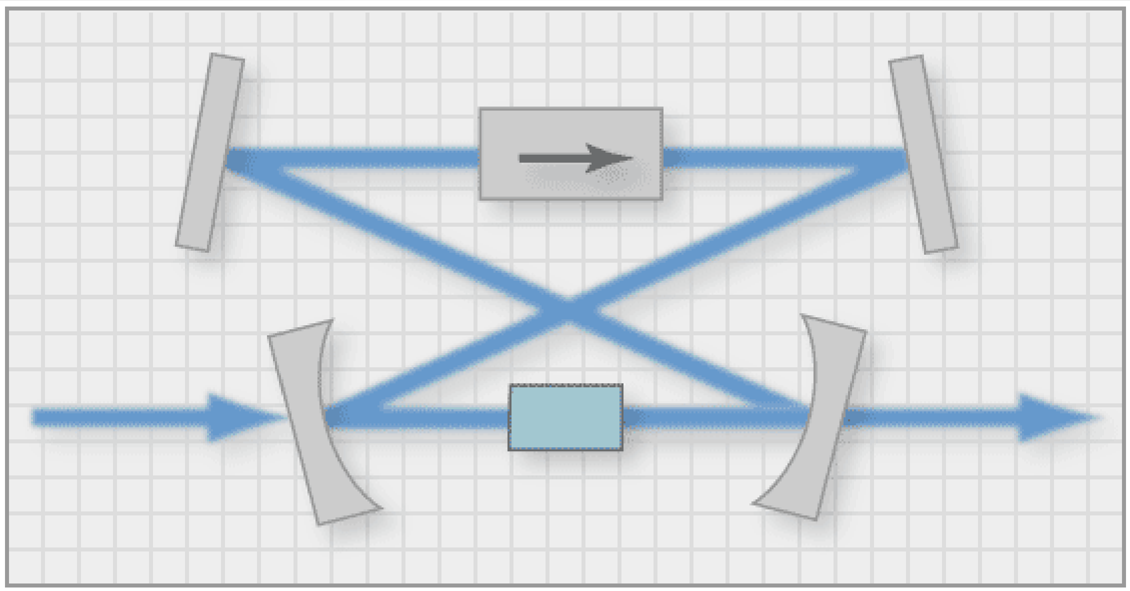
RP Resonator V4



a software product of

RP Photonics Consulting GmbH

www.rp-photonics.com/resonator.html

RP PHOTONICS

Why is Resonator Design so Important?

- Beam radius in laser crystal is essential for **power conversion** and **beam quality** of the laser output.
- Beam radius is affected by **thermal lensing** – resonator must work well within some range of dioptric powers.
- Output beam quality may be spoiled by **mode coupling effects**.
- **Alignment sensitivity** depends critically on the resonator design!
- Design changes affect all sorts of resonator properties – **cannot fix problems step by step!** Need to find a design which is good in *all* respects.

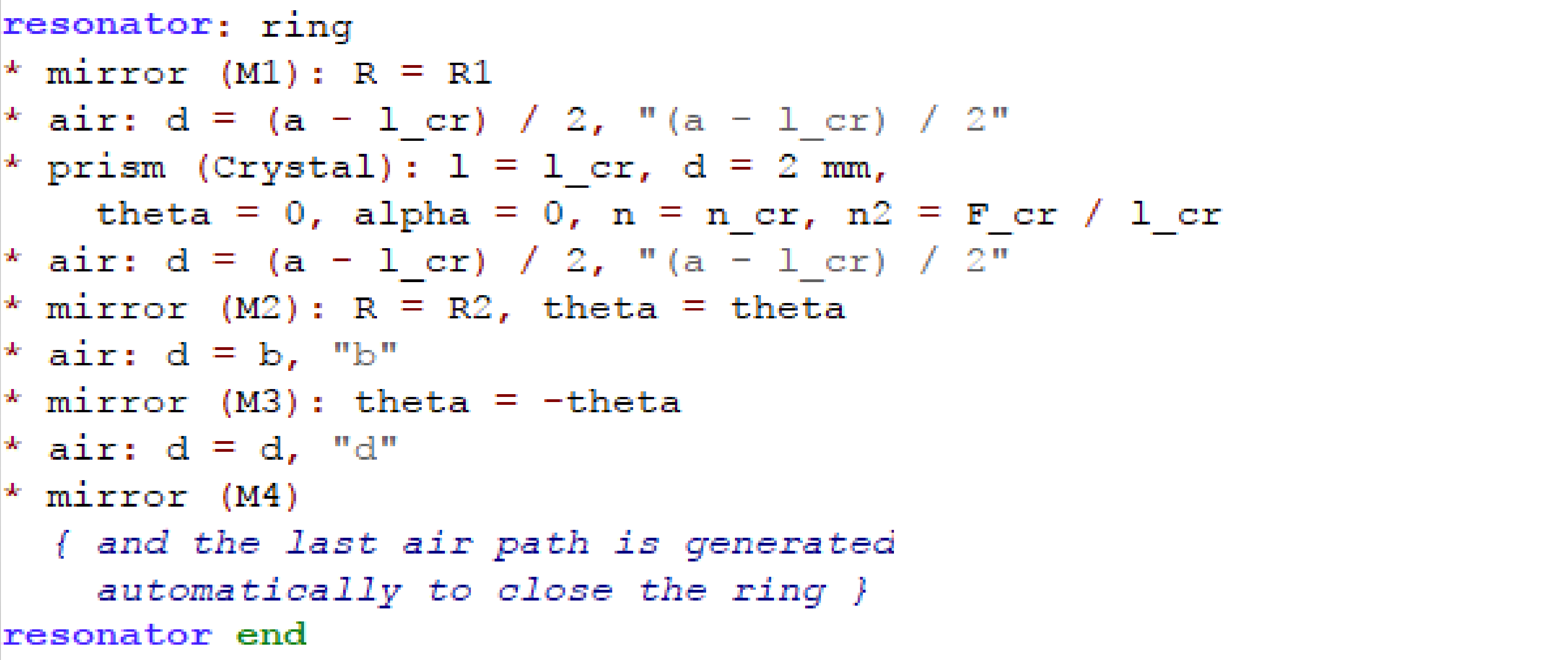
- **A properly designed laser resonator is essential for laser performance!**
- **The design task is non-trivial; you need powerful software *and* highly competent technical support.**

What is Special about the **RP Resonator** software?

- You can define resonator structures, additional calculations or optimizations, graphical diagrams etc. in text form – i.e., as **script code**.
- This approach is **far more flexible** than working with forms or pop-up menus:
  - can easily **parametrize designs**: all arm lengths, for example, are automatically calculated from a few input parameters
  - can define multiple resonator arms with a simple loop
  - can **freely define detailed optimization goals** in the form of a figure-of-merit function (not just enter parameters of a given function!)
  - can **define your own diagrams**, containing any curves and additional elements – no limitation to predefined types of diagrams!
- **Note: flexibility is not just nice to have, but essential for sophisticated design tasks!**

Scripting is Easy!

Example 1: definition of a bow-tie ring resonator:

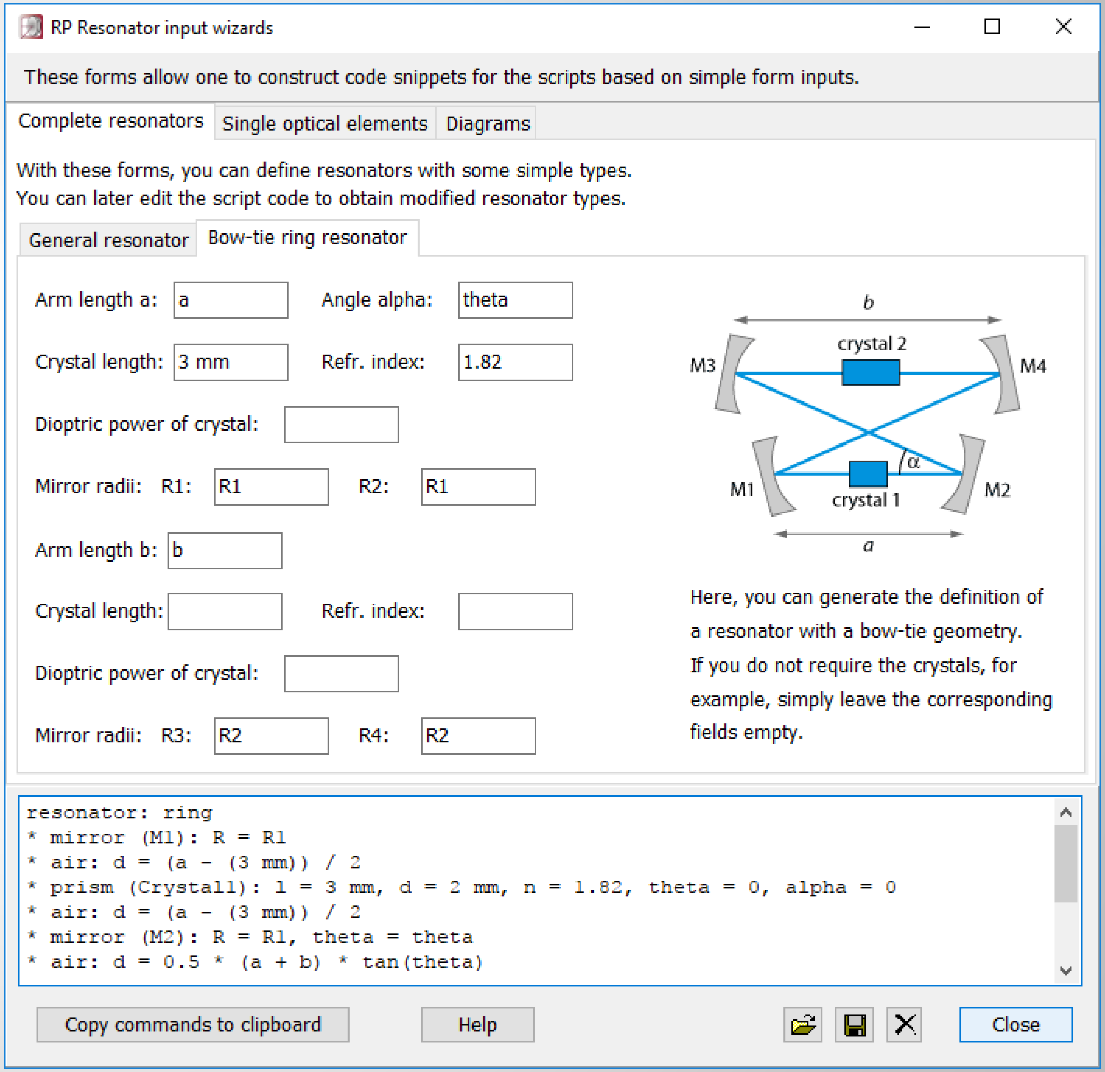


```
resonator: ring
* mirror (M1): R = R1
* air: d = (a - l_cr) / 2, "(a - l_cr) / 2"
* prism (Crystal): l = l_cr, d = 2 mm,
    theta = 0, alpha = 0, n = n_cr, n2 = F_cr / l_cr
* air: d = (a - l_cr) / 2, "(a - l_cr) / 2"
* mirror (M2): R = R2, theta = theta
* air: d = b, "b"
* mirror (M3): theta = -theta
* air: d = d, "d"
* mirror (M4)
  { and the last air path is generated
    automatically to close the ring }
resonator end
```

(Design depends on a few parameters like a, b, d and theta,
which have previously defined – not shown here.)

Scripting is Easy!

Script code for common types of resonators can also be generated with an interactive form – just enter the parameters and copy the generated script code.



These forms allow one to construct code snippets for the scripts based on simple form inputs.

With these forms, you can define resonators with some simple types.
You can later edit the script code to obtain modified resonator types.

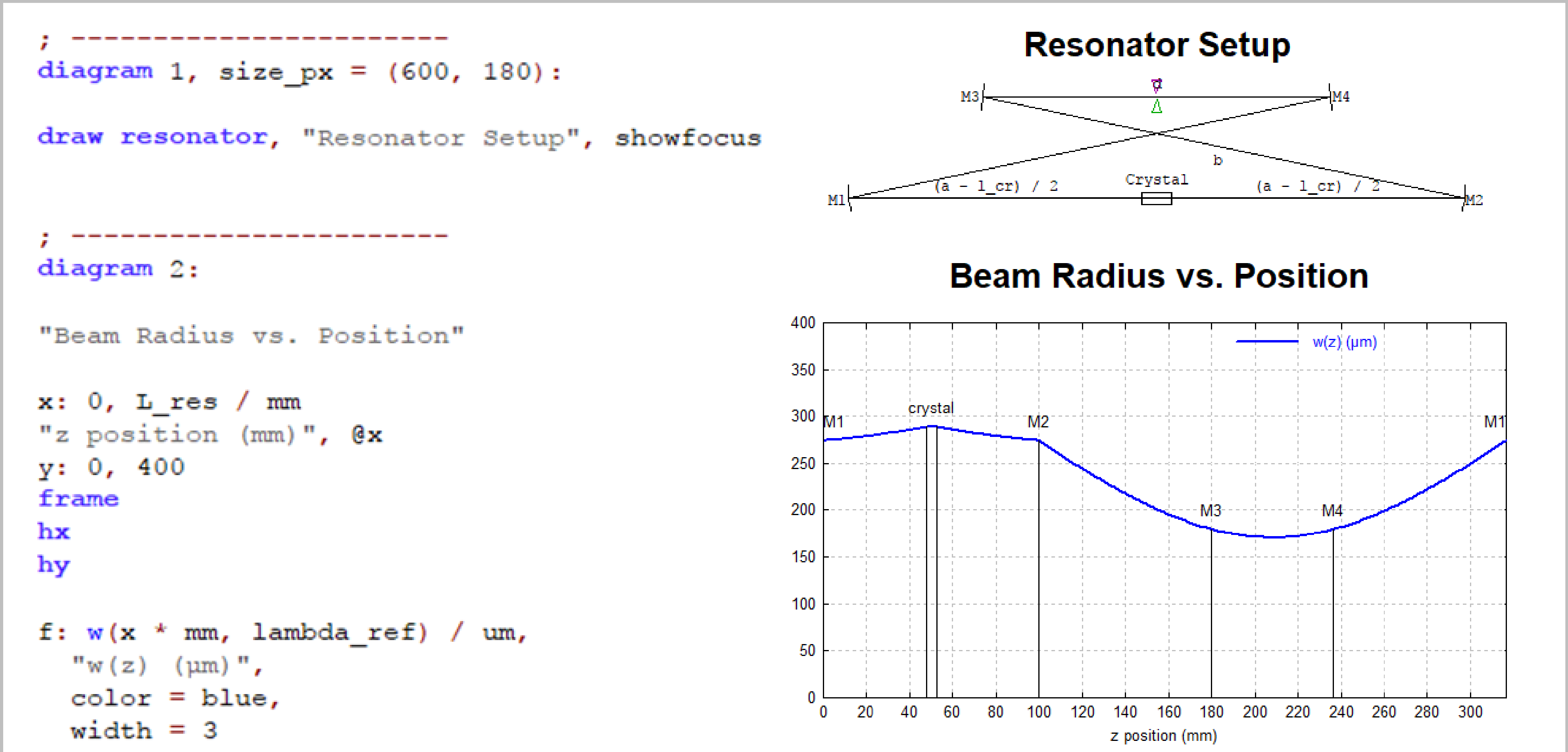
Arm length a: a
Angle alpha: theta
Crystal length: 3 mm
Refr. index: 1.82
Dioptric power of crystal:
Mirror radii: R1: R1 R2: R1
Arm length b: b
Crystal length:
Refr. index:
Dioptric power of crystal:
Mirror radii: R3: R2 R4: R2

Here, you can generate the definition of a resonator with a bow-tie geometry. If you do not require the crystals, for example, simply leave the corresponding fields empty.

```
resonator: ring
* mirror (M1): R = R1
* air: d = (a - (3 mm)) / 2
* prism (Crystal1): l = 3 mm, d = 2 mm, n = 1.82, theta = 0, alpha = 0
* air: d = (a - (3 mm)) / 2
* mirror (M2): R = R1, theta = theta
* air: d = 0.5 * (a + b) * tan(theta)
```

Scripting is Easy!

Example 2: get the resonator displayed and the mode radius plotted:



```
; -----------------------
diagram 1, size_px = (600, 180):

draw resonator, "Resonator Setup", showfocus


; -----------------------
diagram 2:

"Beam Radius vs. Position"

x: 0, L_res / mm
"z position (mm)", @x
y: 0, 400
frame
hx
hy

f: w(x * mm, lambda_ref) / um,
  "w(z) (µm)",
  color = blue,
  width = 3
```

Scripting is Enormously Flexible!

Many tasks can easily be accomplished with a few lines of script code – for example:

- Generate tailored **graphical diagrams** for visualizing properties of your resonator or whatever else.
- Define a **figure-of-merit function** as a precise definition of your optimization target, and do a numerical optimization of some resonator parameters such that the value of that function becomes minimal.
- Save any calculated data in a **text file or binary file** – essentially any file format can be generated.

You don't depend on which details the software developer has anticipated: put together yourself what you need! You can even do full-blown programming for most sophisticated calculations.

How to Get Scripts Developed?

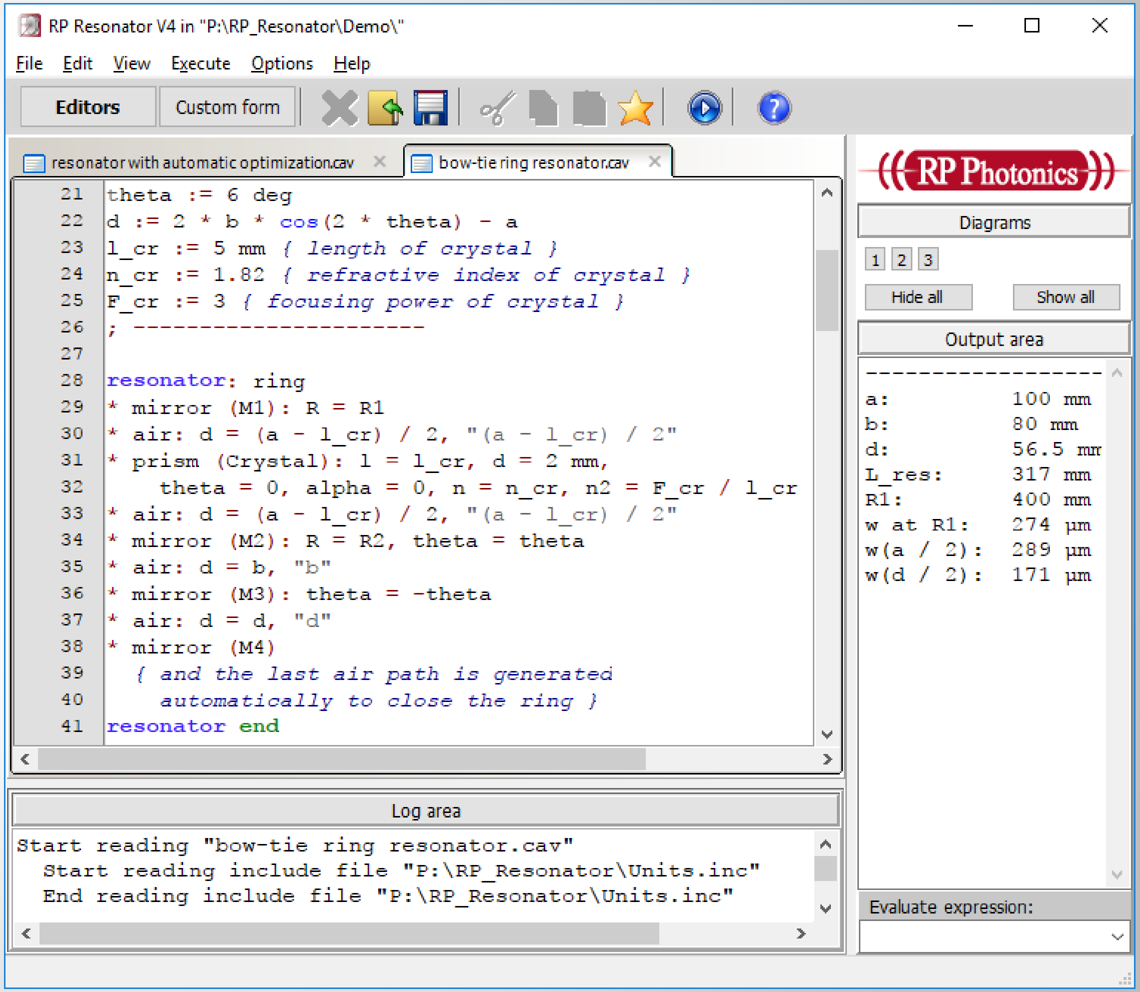
There are different approaches:

- Copy one of the **demo scripts** and modify it according to your needs.
- Adapt a **previously developed script** to the new requirements.
- Use the **code snippets library** for getting frequently used parts of script code. (Also add your own code snippets to that!)
- Get help within the **technical support**. Describe your needs, and we send you a script as a starting point for your development.

The User Interface (1)

Powerful script editors and editing tools:

- **Wizard for resonator definitions**
- **Code snippet library** for frequently used parts of code
- **Parameter hints** for predefined functions
- **Multilevel undo/redo**
- **Syntax highlighting** for good readability of code
- **Integrated syntax checker**
- **Automatic code formatting** for consistent formats
- Setting of **breakpoints** for easy debugging



```
theta := 6 deg
d := 2 * b * cos(2 * theta) - a
l_cr := 5 mm { length of crystal }
n_cr := 1.82 { refractive index of crystal }
F_cr := 3 { focusing power of crystal }
; ----------------------

resonator: ring
* mirror (M1): R = R1
* air: d = (a - l_cr) / 2, "(a - l_cr) / 2"
* prism (Crystal): l = l_cr, d = 2 mm,
    theta = 0, alpha = 0, n = n_cr, n2 = F_cr / l_cr
* air: d = (a - l_cr) / 2, "(a - l_cr) / 2"
* mirror (M2): R = R2, theta = theta
* air: d = b, "b"
* mirror (M3): theta = -theta
* air: d = d, "d"
* mirror (M4)
  { and the last air path is generated
    automatically to close the ring }
resonator end
```

```
--------------------
a:          100 mm
b:          80 mm
d:          56.5 mm
L_res:      317 mm
R1:         400 mm
w at R1:    274 µm
w(a / 2):   289 µm
w(d / 2):   171 µm
```

```
Start reading "bow-tie ring resonator.cav"
  Start reading include file "P:\RP_Resonator\Units.inc"
  End reading include file "P:\RP_Resonator\Units.inc"
```

The User Interface (2)

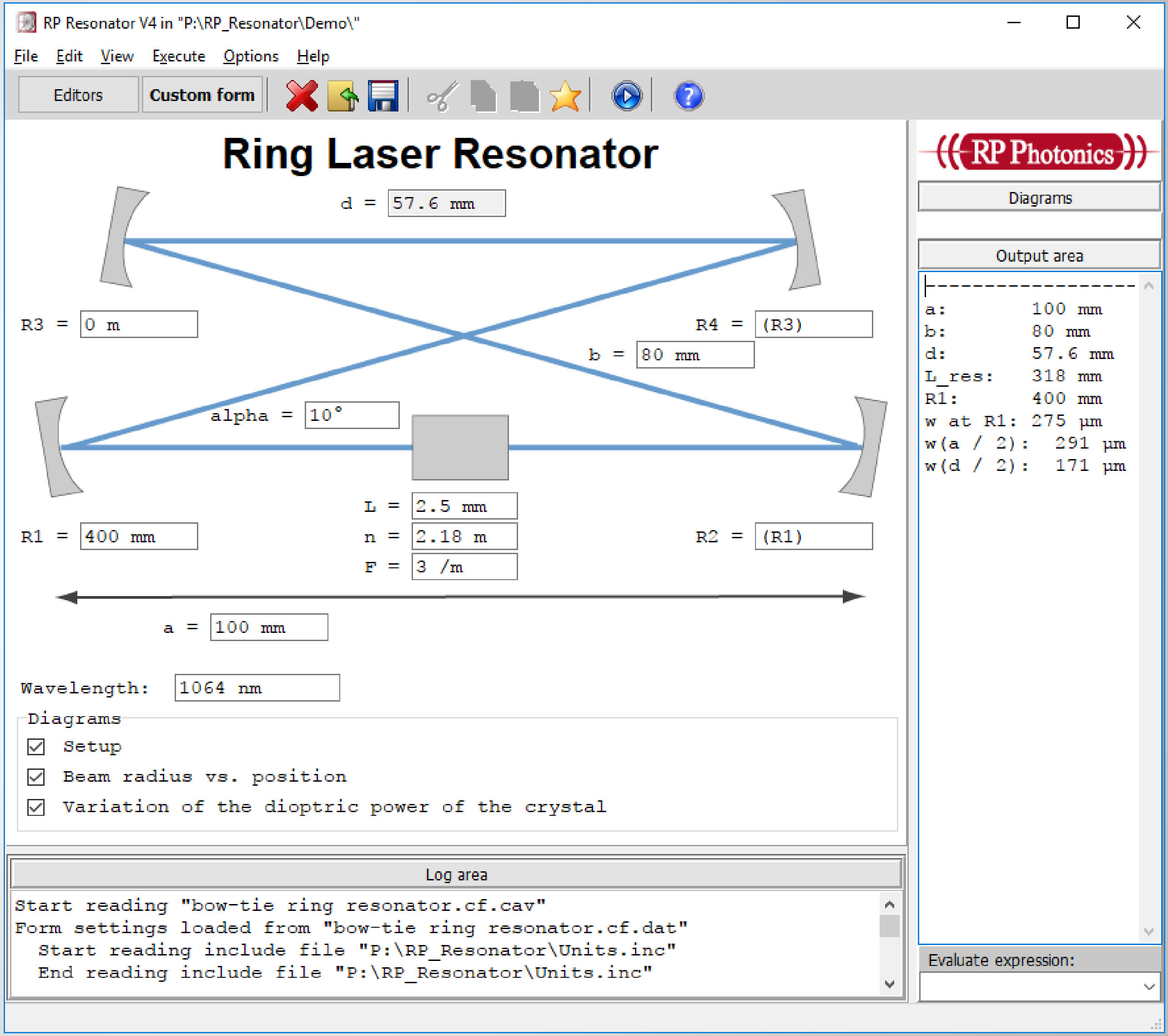
Custom forms: get any tailored forms you need!

- Such forms **can be made for *any* simulation**!
- **Very easy to use:** just fill out the input fields and execute to see the output values as well as created graphical diagrams.
  (See the example on the next page.)
- You can either **make such forms yourself or get them made** within the technical support. (A custom form is defined quite simply in text form within a script.)
- Ideal combination of flexibility and ease of use!
- Consequently, **RP Resonator** becomes more suitable also for those who need to get certain designs recalculated **without spending much time on technical details**.

The User Interface (3)

Simple example for custom forms:

ring laser resonator, calculating the resonator modes from arm lengths, mirror curvatures etc., also generating various diagrams



Ring Laser Resonator

d = 57.6 mm

R3 = 0 m

R4 = (R3)

b = 80 mm

alpha = 10°

L = 2.5 mm

n = 2.18 m

F = 3 /m

R1 = 400 mm

R2 = (R1)

a = 100 mm

Wavelength: 1064 nm

Diagrams

☑ Setup

☑ Beam radius vs. position

☑ Variation of the dioptric power of the crystal

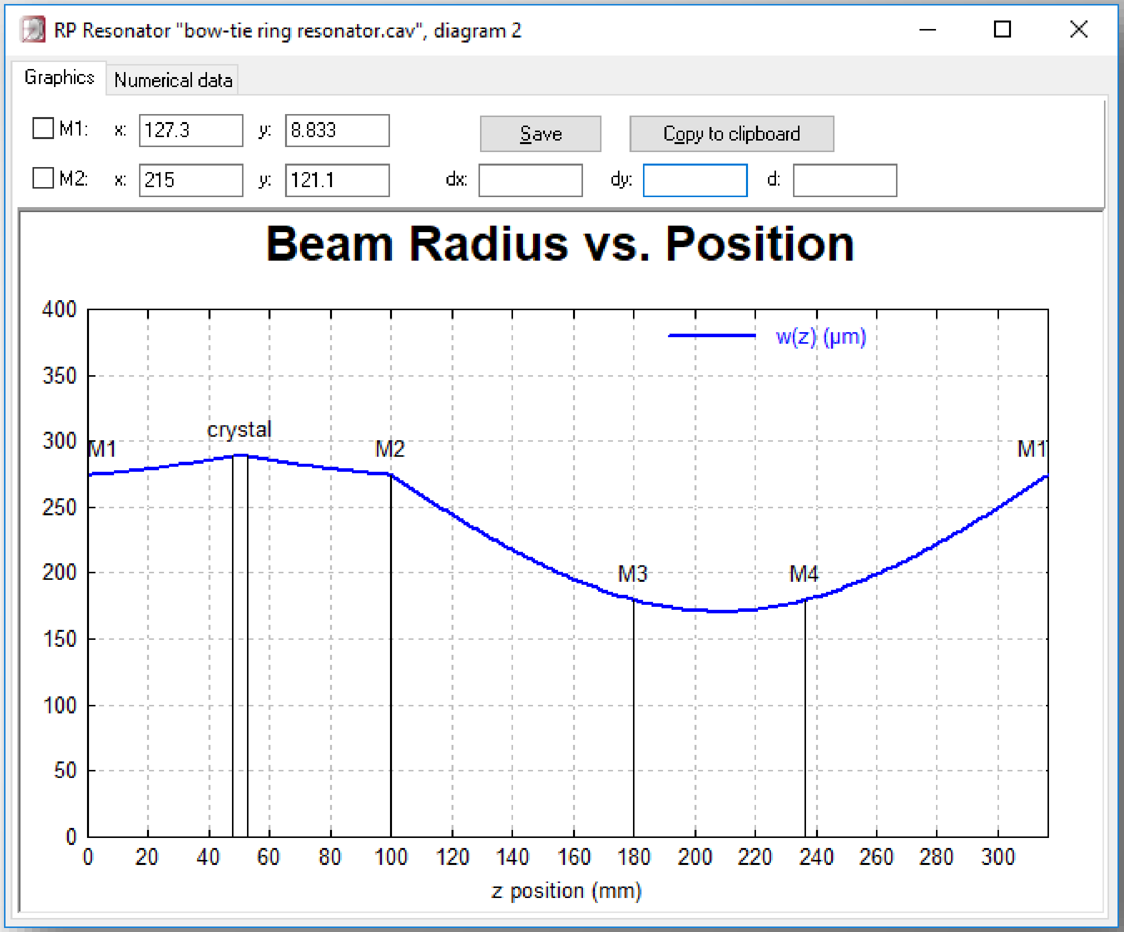
```
a:        100 mm
b:        80 mm
d:        57.6 mm
L_res:    318 mm
R1:       400 mm
w at R1: 275 µm
w(a / 2):  291 µm
w(d / 2):  171 µm
```

```
Start reading "bow-tie ring resonator.cf.cav"
Form settings loaded from "bow-tie ring resonator.cf.dat"
  Start reading include file "P:\RP_Resonator\Units.inc"
  End reading include file "P:\RP_Resonator\Units.inc"
```

The User Interface (4)

Graphical output windows

- high-quality graphics, directly usable for publications:
  copy to clipboard or save to file
- can make animated graphics
- adjustable resolution
- markers for doing measurements
- export numerical data



RP Resonator "bow-tie ring resonator.cav", diagram 2

□ M1: x: 127.3 y: 8.833

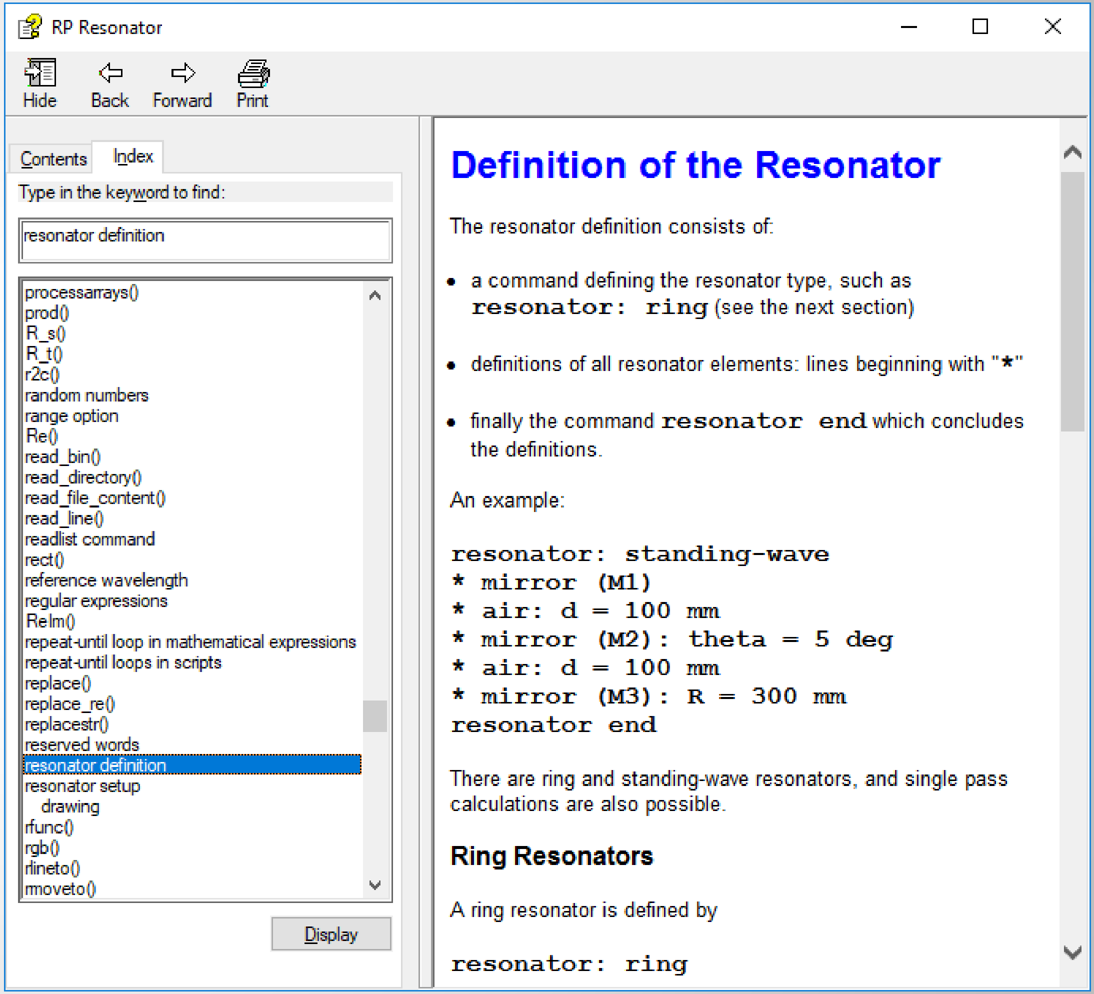
□ M2: x: 215 y: 121.1 dx: dy: d:

Also have flexible options for generating output in text form!

Put that into diagrams or files as you like.

Documentation

- comprehensive **PDF manual**
- detailed **online help system**
- comprehensive explanations of the used physical model, details of the script language, etc.
- various **demo files**, demonstrating many different possibilities



Type in the keyword to find:

resonator definition

processarrays()
prod()
R_s()
R_t()
r2c()
random numbers
range option
Re()
read_bin()
read_directory()
read_file_content()
read_line()
readlist command
rect()
reference wavelength
regular expressions
ReIm()
repeat-until loop in mathematical expressions
repeat-until loops in scripts
replace()
replace_re()
replacestr()
reserved words
resonator definition
resonator setup
drawing
rfunc()
rgb()
rlineto()
rmoveto()

Definition of the Resonator

The resonator definition consists of:

- a command defining the resonator type, such as `resonator: ring` (see the next section)
- definitions of all resonator elements: lines beginning with "*"
- finally the command `resonator end` which concludes the definitions.

An example:

```
resonator: standing-wave
* mirror (M1)
* air: d = 100 mm
* mirror (M2): theta = 5 deg
* air: d = 100 mm
* mirror (M3): R = 300 mm
resonator end
```

There are ring and standing-wave resonators, and single pass calculations are also possible.

Ring Resonators

A ring resonator is defined by

```
resonator: ring
```

Technical Support

Any remaining technical issues can be addressed with the technical support:

The price for a **commercial user license** contains **8 support hours** (non-commercial licenses: 4 hours).

The support is done by Dr. Paschotta himself, who is a distinguished expert in this area and has developed **RP Resonator**. He will make sure that you become another very satisfied user of the software!

Dr. Rüdiger Paschotta,
founder and managing director
of RP Photonics,
developer of RP Resonator

Note that RP Photonics also offers consultancy on laser technology.

Can I Afford This Software?

Sure, a high-quality software product including competent support from a top expert costs some money.

Anyway, the better question is:

Can I afford *not* to have a powerful software tool, i.e.,

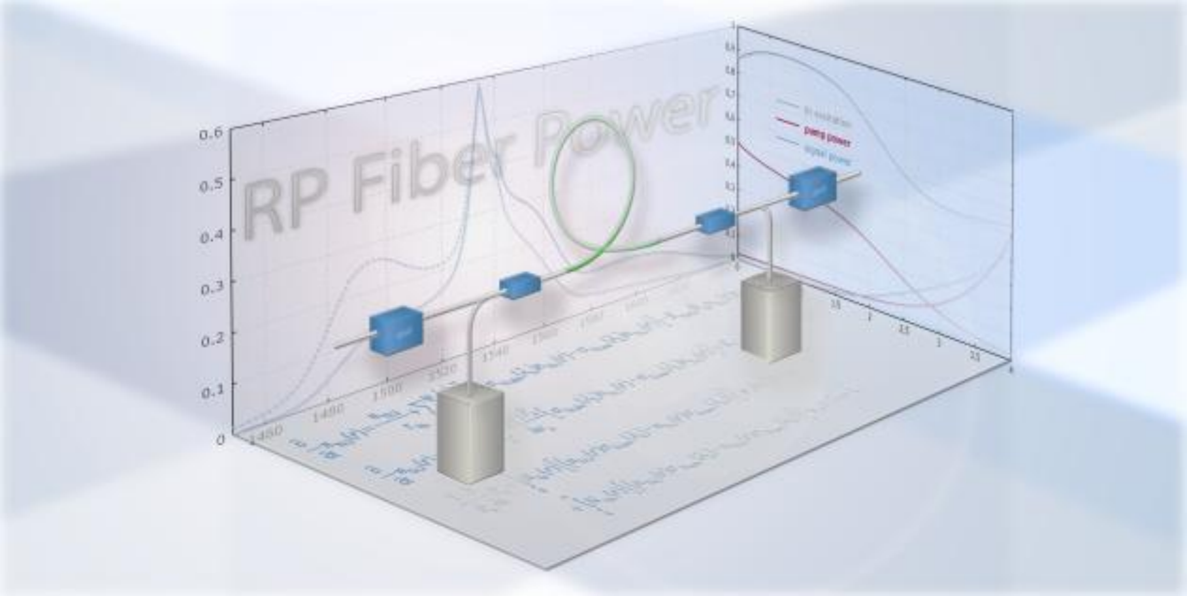
- to muddle through with insufficient tools?
- to use trial & error, wasting time and materials?
- to let customers wait while my competitors sell their products?

The **RP Resonator** software will give a boost to your productivity! Also, your employees or students will become productive sooner when they acquire a deep understanding by playing with this software.

Other Software from RP Photonics

RP Fiber Power:

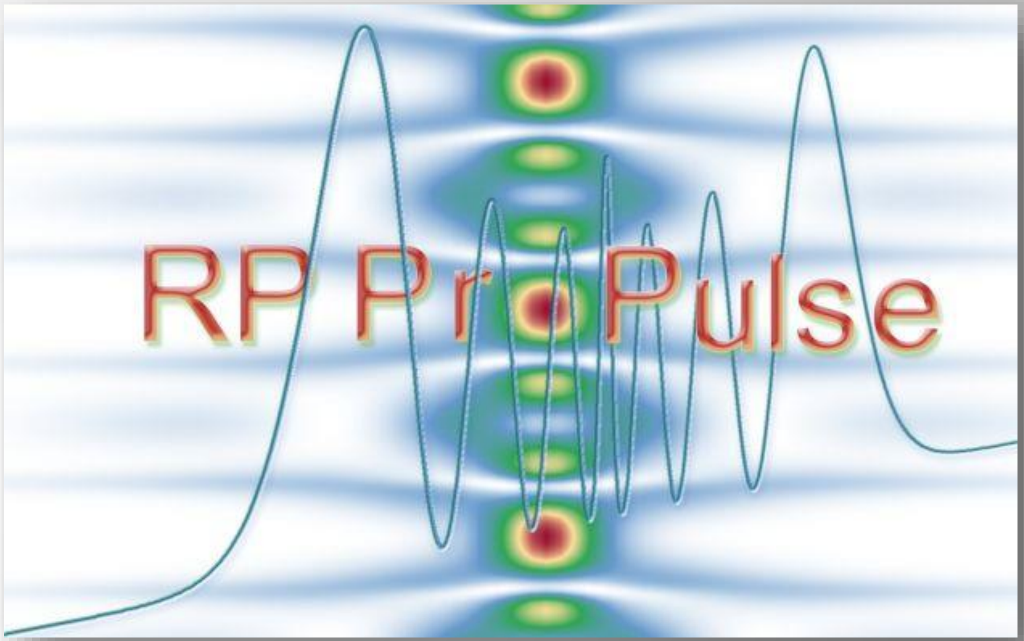
- design of fiber amplifiers, fiber amplifiers, double-clad fibers, multi-core fibers, fiber couplers, etc.
- powerful script language for an enormous flexibility
- can do most sophisticated analysis and optimizations



See a detailed description: www.rp-photonics.com/fiberpower.html

Other Software from RP Photonics

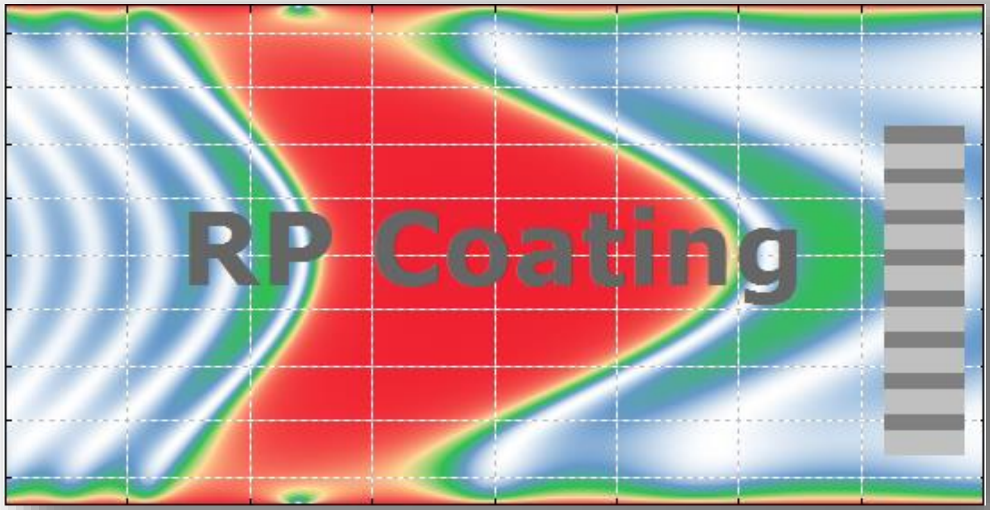
RP ProPulse:



- simulates the propagation of ultrashort pulses e.g. in mode-locked lasers or sync-pumped OPOs
- can include laser gain, parametric gain, SHG, Kerr and Raman effect, chromatic dispersion, etc.
- pulse display window
- can do most sophisticated analysis and optimizations

See a detailed description: www.rp-photonics.com/propulse.html

Other Software from RP Photonics



RP Coating:

- analysis of multilayer thin-film devices: laser mirrors, filters, anti-reflection coatings, dispersive mirrors, polarizers, SESAMs, VECSELs, …
- can fully parameterize designs
- read / write data from or to text files or binary files with arbitrary formats:
  read transmission spectra from a spectrometer, control a coating machine, etc.
- can do most sophisticated analysis and optimizations

See a detailed description: www.rp-photonics.com/coating.html