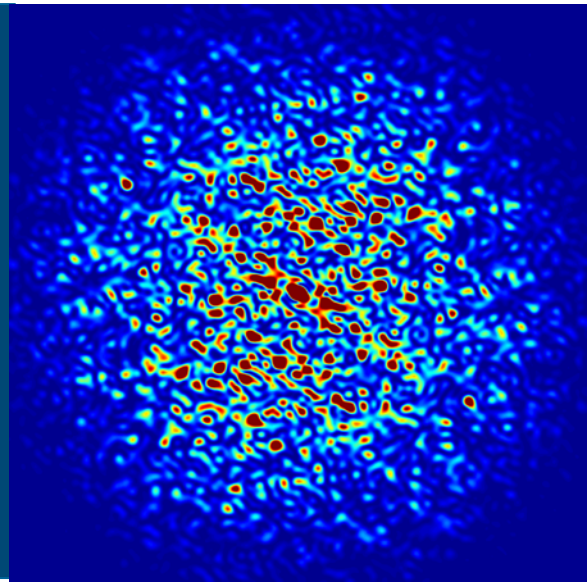
Introduction to OASYS-SRW



Luca Rebuffi (ANL)

Second OASYS School
APS-ANL, Lemont, IL

December 11-13, 2019

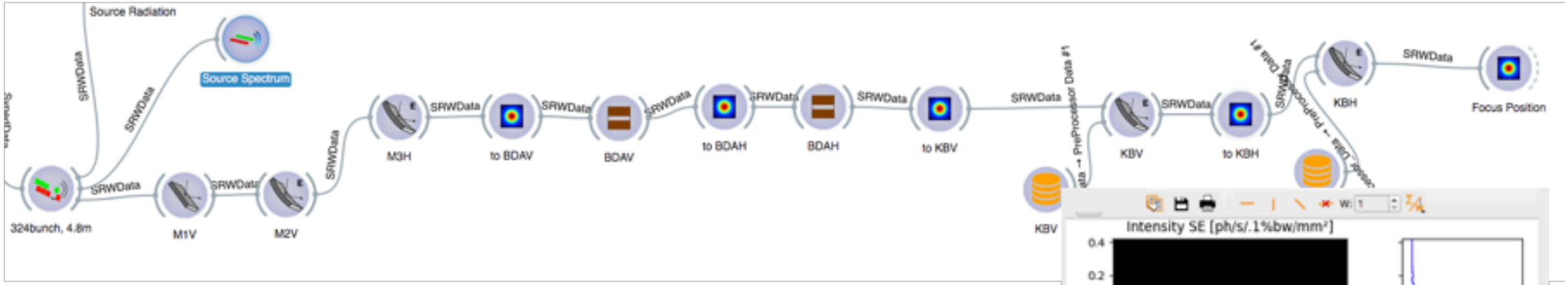
Outline

- Introduction
- OASYS Automations
- Propagation modes
- Multi-Electron tools

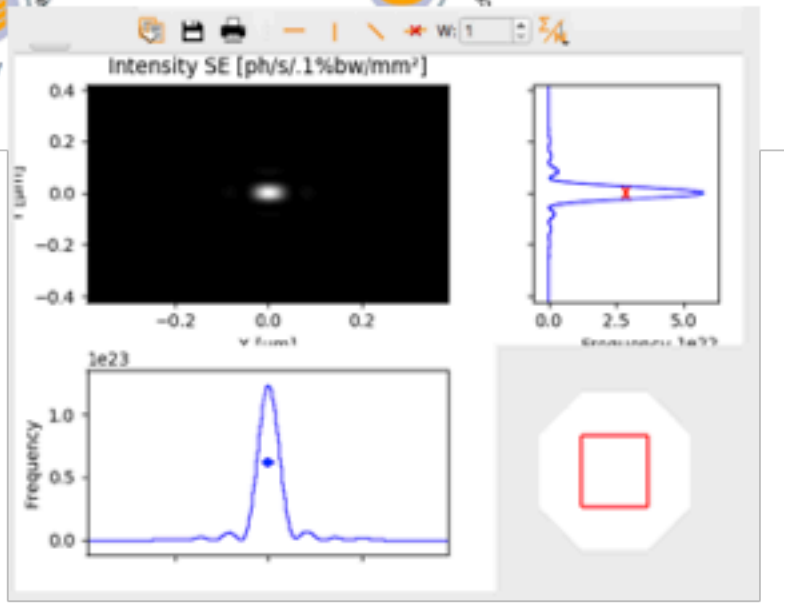
INTRODUCTION

Introduction to OASYS-SRW

OASYS (OrAnge SYnchrotron Suite) integrates SRW (Synchrotron Radiation Workshop) by Oleg Chubar (BNL), by providing it a Graphic User Interface and tools to reproduce its native usage.



OASYS-SRW is fully integrated with SYNED and WOFRY frameworks and XOPPY, XrayServer, DABAM tools

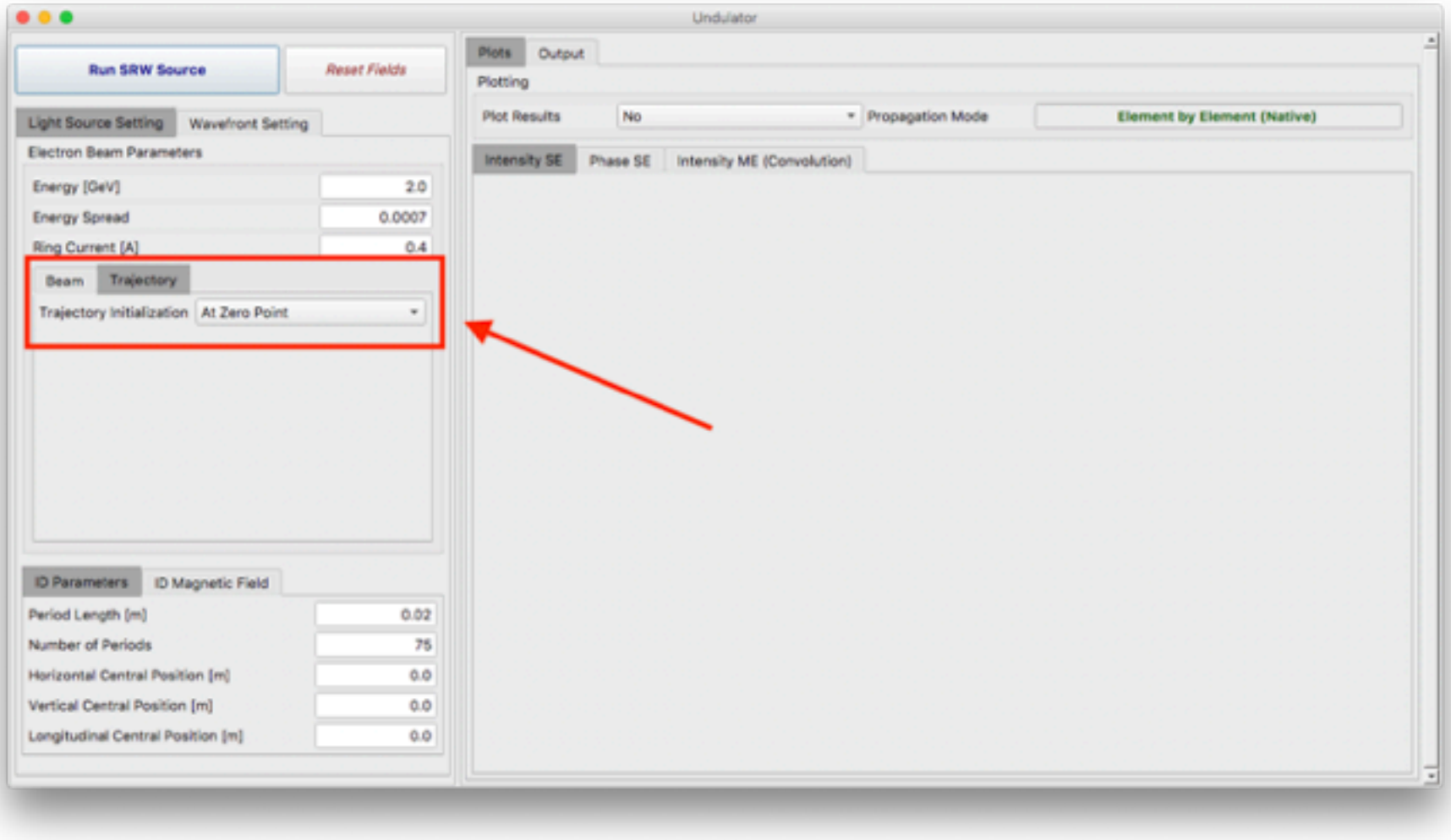


https://github.com/ochubar/SRW

OASYS AUTOMATIONS

OASYS Automations: Sources

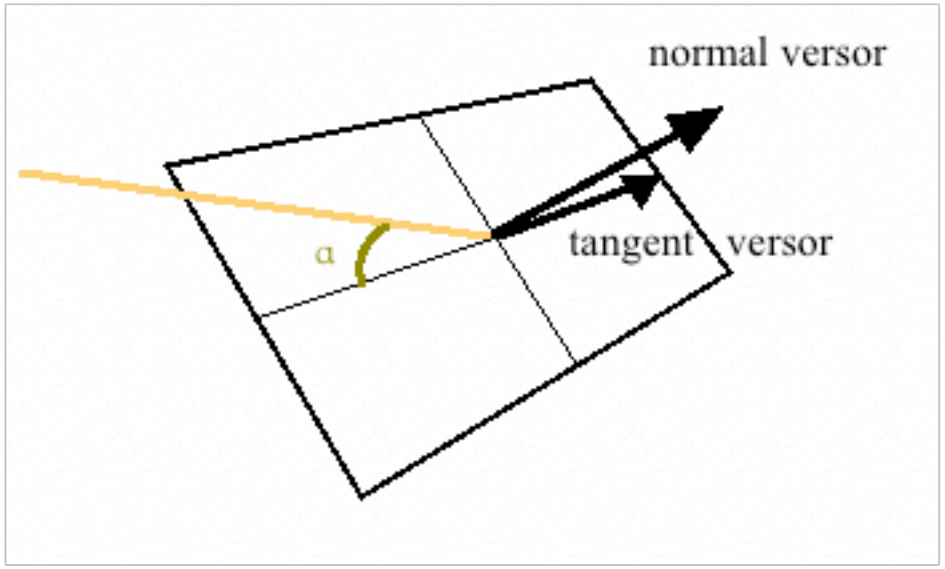
Trajectory calculation should begin before the ID. A correct setup is given by the option "At Zero Point", where the z^0 (longitudinal initial position of the electron) value is computed automatically. The "Sampled from phase space" option make sense only for loops, providing a "visual" multielectron simulation inside OASYS.



OASYS Automations: Optics Elements

The orientation of the optical elements in SRW is given by the components of two versors: normal to the surface and tangent to the surface.

Rules are defined in the following, where α is the GRAZING angle (that is $90° - \theta_i$, where θ_i is the incident angle in the input form):



Orientation: LEFT

$$nv_x = -\cos\alpha,\ nv_y = 0.0,\ nv_z = -\sin\alpha$$
$$tv_x = \pm\sin\alpha,\ tv_y = 0.0$$

Orientation: RIGHT

$$nv_x = \cos\alpha,\ nv_y = 0.0,\ nv_z = -\sin\alpha$$
$$tv_x = \mp\sin\alpha,\ tv_y = 0.0$$

Orientation: UP

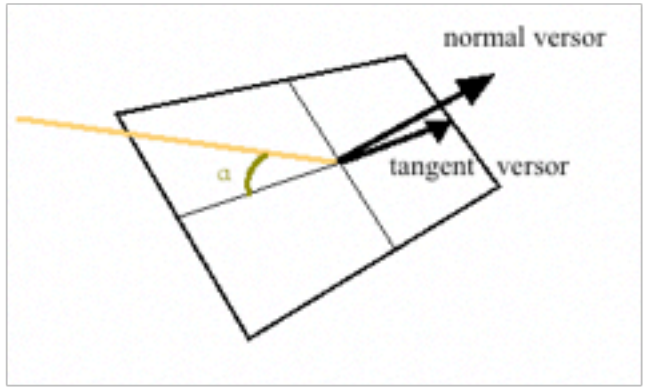
$$nv_x = 0.0,\ nv_y = \cos\alpha,\ nv_z = -\sin\alpha$$
$$tv_x = 0.0,\ tv_y = \mp\sin\alpha$$

Orientation: DOWN

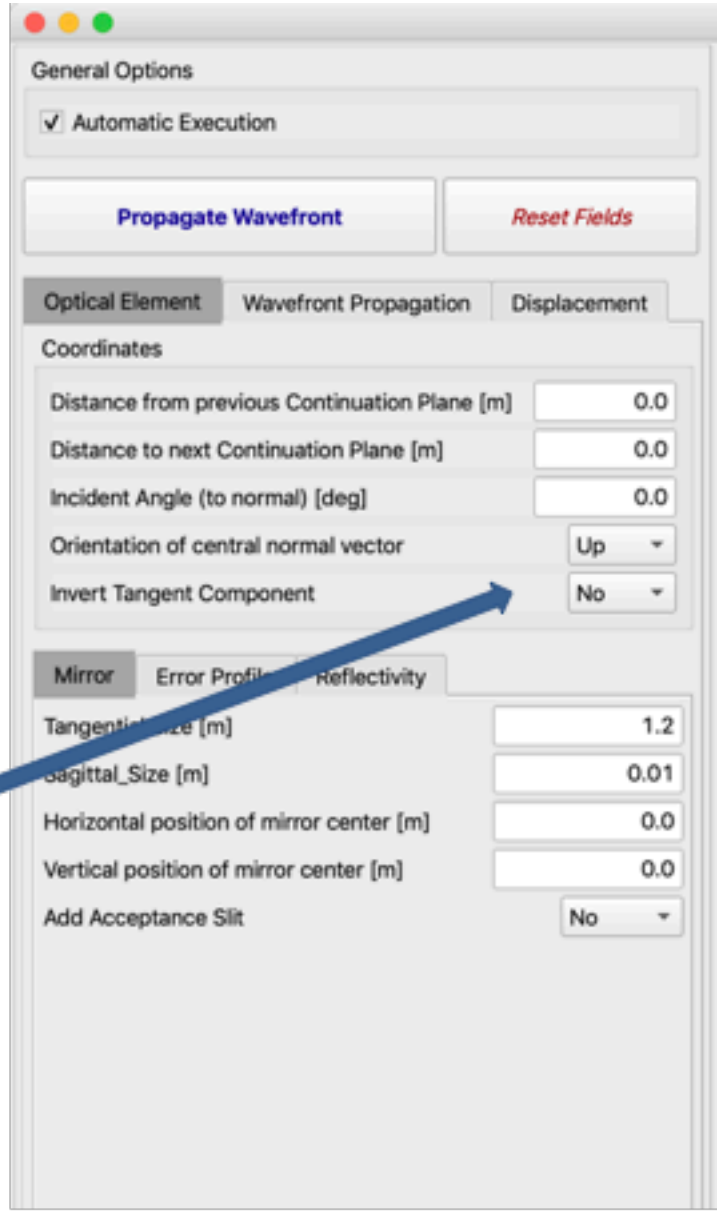
$$nv_x = 0.0,\ nv_y = -\cos\alpha,\ nv_z = -\sin\alpha$$
$$tv_x = 0.0,\ tv_y = \pm\sin\alpha$$

OASYS Automations: Optics Elements

OASYS provides an automatic calculation of these components, and the tangential component could be easily inverted by a dedicated option.
In principle, tangential components sign doesn't really matter for mirrors. In the case of gratings and crystals, there may be a difference however: these grating and crystal cases may need additional tests to debug/understand their orientation (e.g. with positive/negative diffraction orders for gratings or by observing shape of rocking curve vs transverse position / angle for crystals). In real life, the correct orientation of the tangential component is easily chosen by a "try and fix" approach.



Plane Mirror



OASYS Automations: Gratings

A special case belongs to VLS gratings, where the outcoming angle is different from the incident angle. The direction of propagation should be rotated accordingly, by an angle that can be calculated:

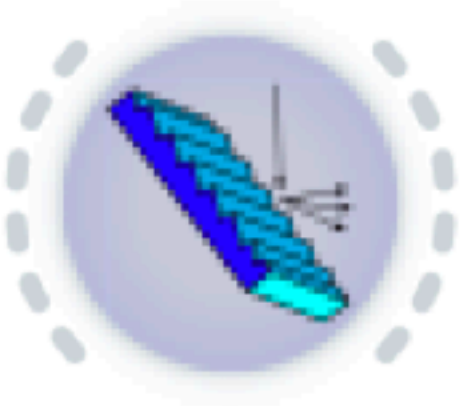
The outcoming angle is

$$\beta = \sin^{-1}(\lambda g_0 - \cos\alpha)$$

Where g_0 is the 0^{th} component of the grooving density.

Then a deflection angle γ is defined as:

$$\gamma = \alpha + \beta + \frac{\pi}{2}$$



Plane Grating

OASYS Automations: Gratings

Orientation: LEFT

$$ooav_x = \sin\gamma, ooav_y = 0.0,\ ooav_z = \cos\gamma$$
$$hbv_x = 0.0,\ hbv_y = \pm 1.0$$

Orientation: RIGHT

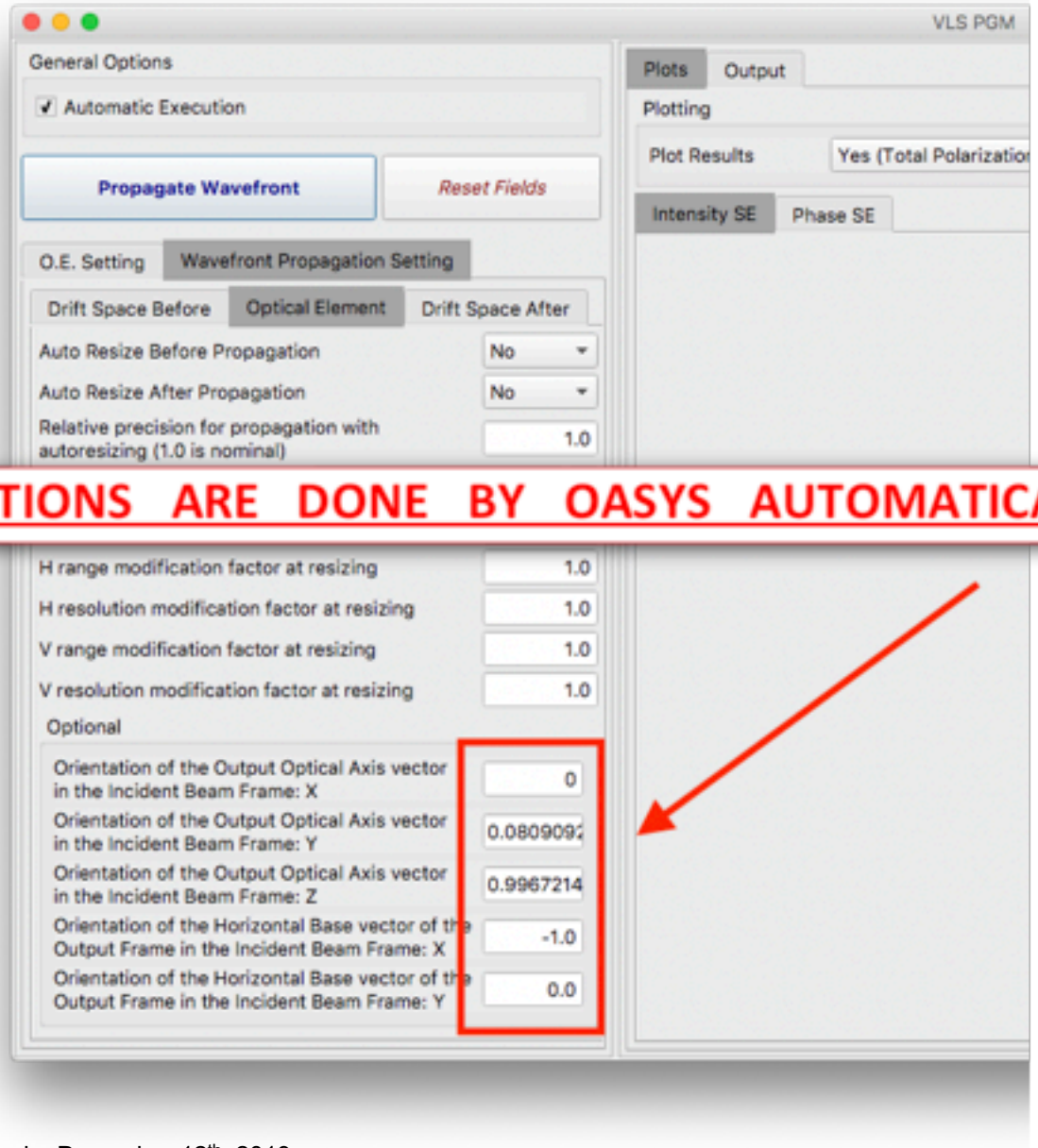
$$ooav_x = -\sin\gamma, ooav_y = 0.0,\ ooav_z = \cos\gamma$$
$$hbv_x = 0.0,\ hbv_y = \pm 1.0$$

Orientation: UP

$$ooav_x = 0.0, ooav_y = \sin\gamma,\ ooav_z = \cos\gamma$$
$$hbv_x = \pm 1.0,\ hbv_y = 0.0$$

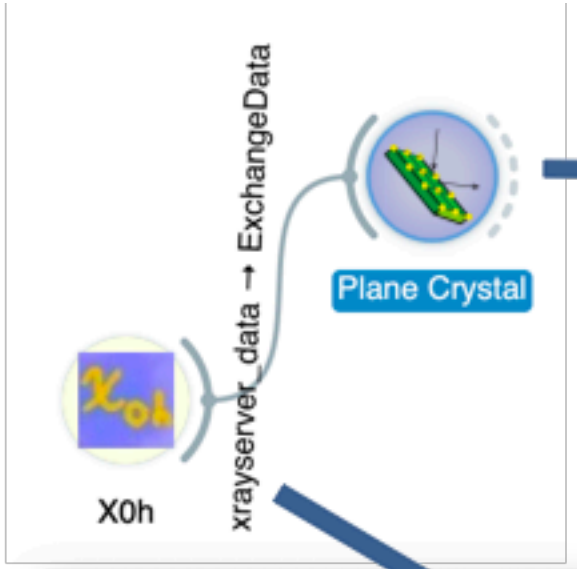
Orientation: DOWN

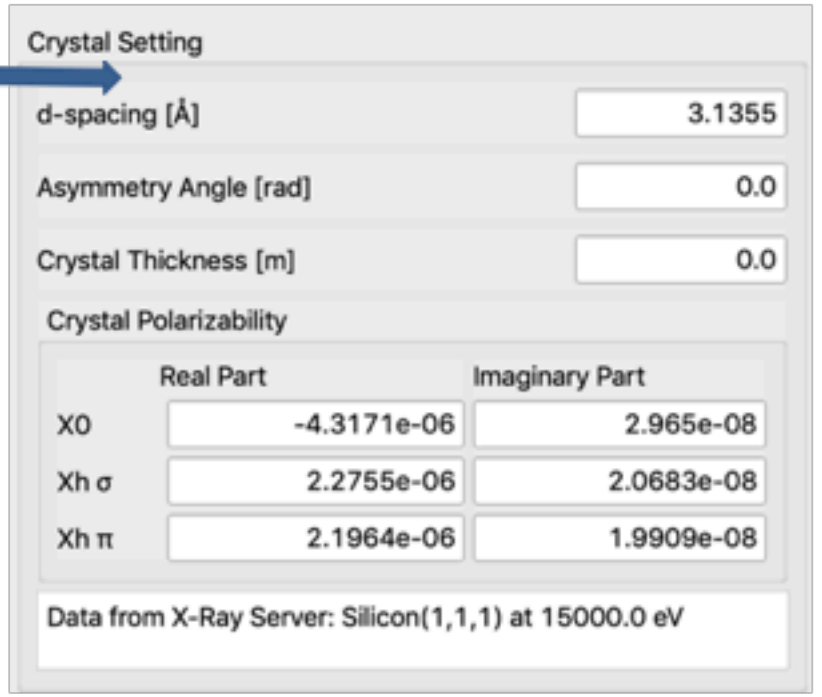
$$ooav_x = 0.0, ooav_y = -\sin\gamma,\ ooav_z = \cos\gamma$$
$$hbv_x = \pm 1.0,\ hbv_y = 0.0$$

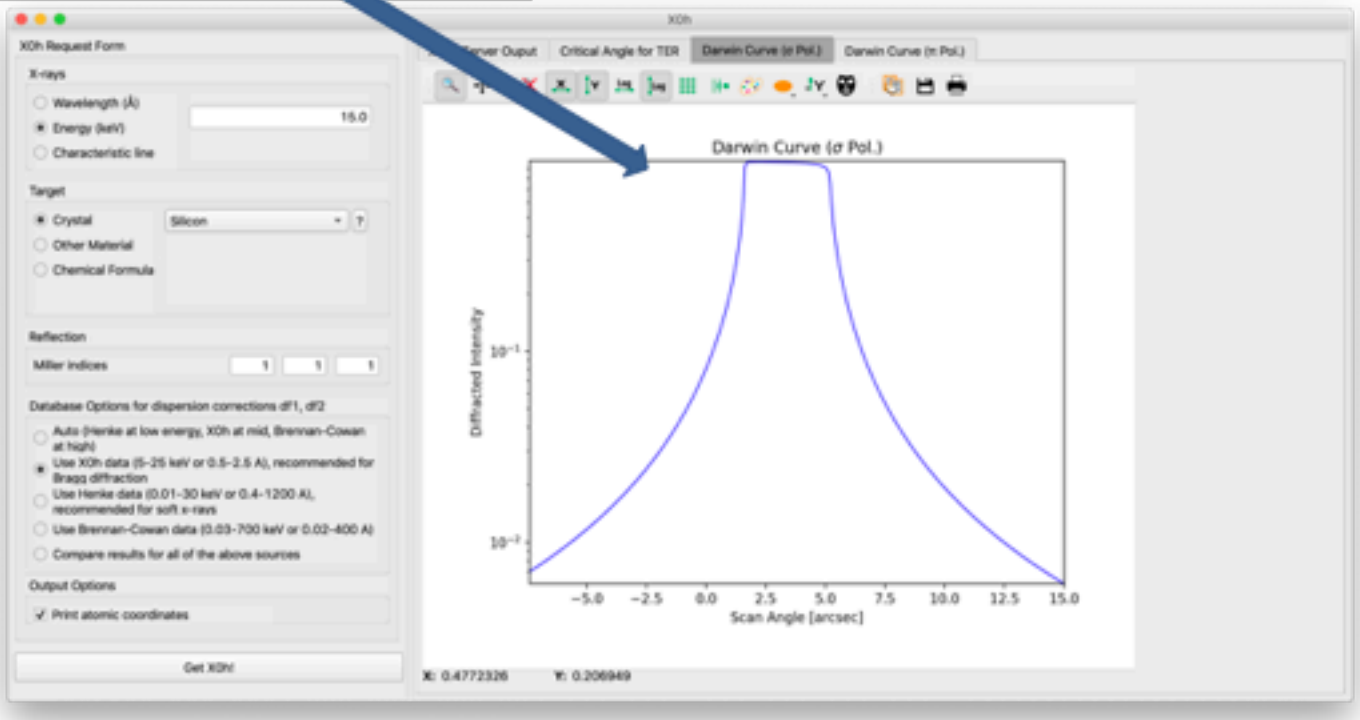


ALL THESE CALCULATIONS ARE DONE BY OASYS AUTOMATICALLY

OASYS Automations: Crystals





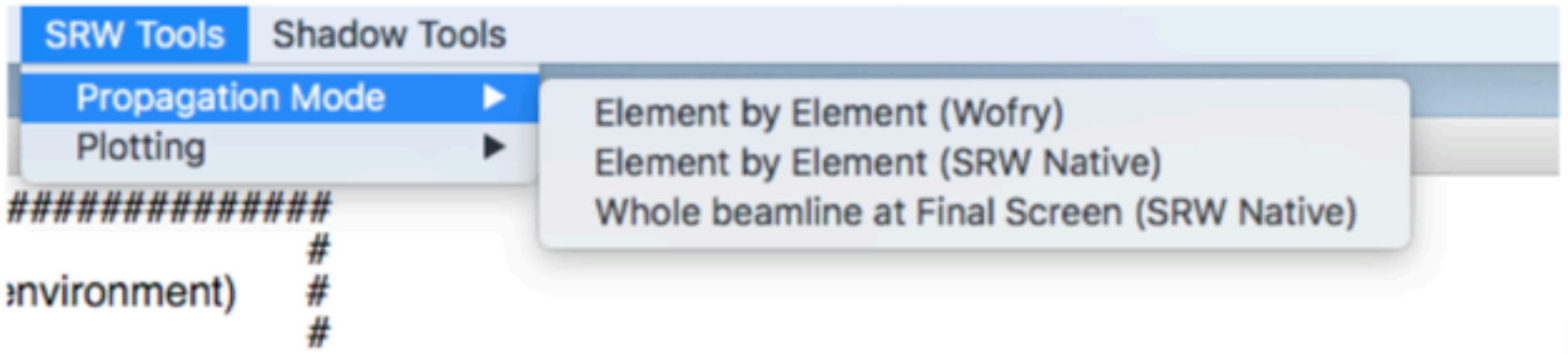


Optical Constants from **XRayServer**

PROPAGATION MODES

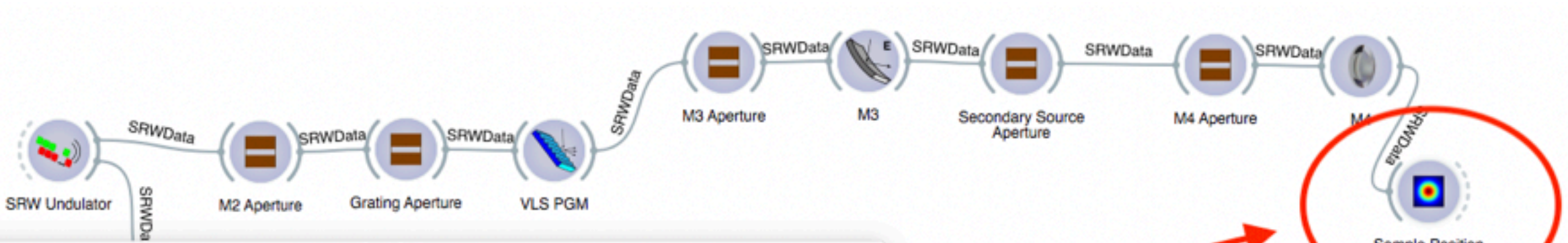
Propagation Modes

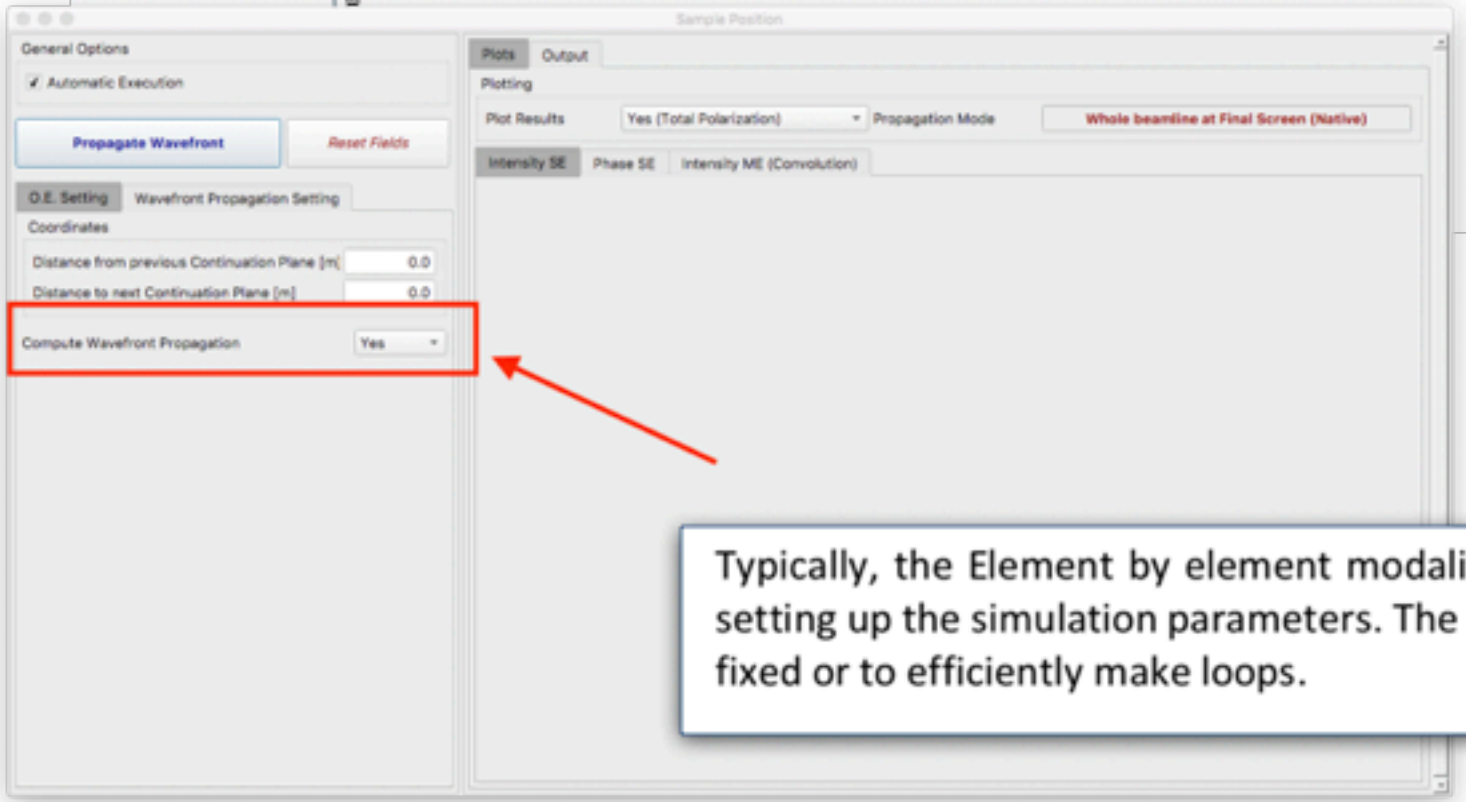
OASYS provides a mechanism to enhance performances during online propagations, by switching on one of the three modalities visible on the SRW Tools Menu:



- Element by Element (Wofry): it uses the general framework for Wavefront Propagations in OASYS. This mechanism is the less efficient for SRW and its use is discouraged.
- Element by Element (SRW Native): it propagates element by element, taking into account the optical element and the associated drift spaces in a single propagation.
- Whole beamline at the Final Screen (SRW Native): it propagates the whole beamline in a single propagation (see Oleg Chubar's Examples distributed with SRW). The propagation is executed in Screen Widgets, where the "Compute Wavefront Propagation" is set on "Yes" (This option is active only within this propagation modality)

Propagation Modes

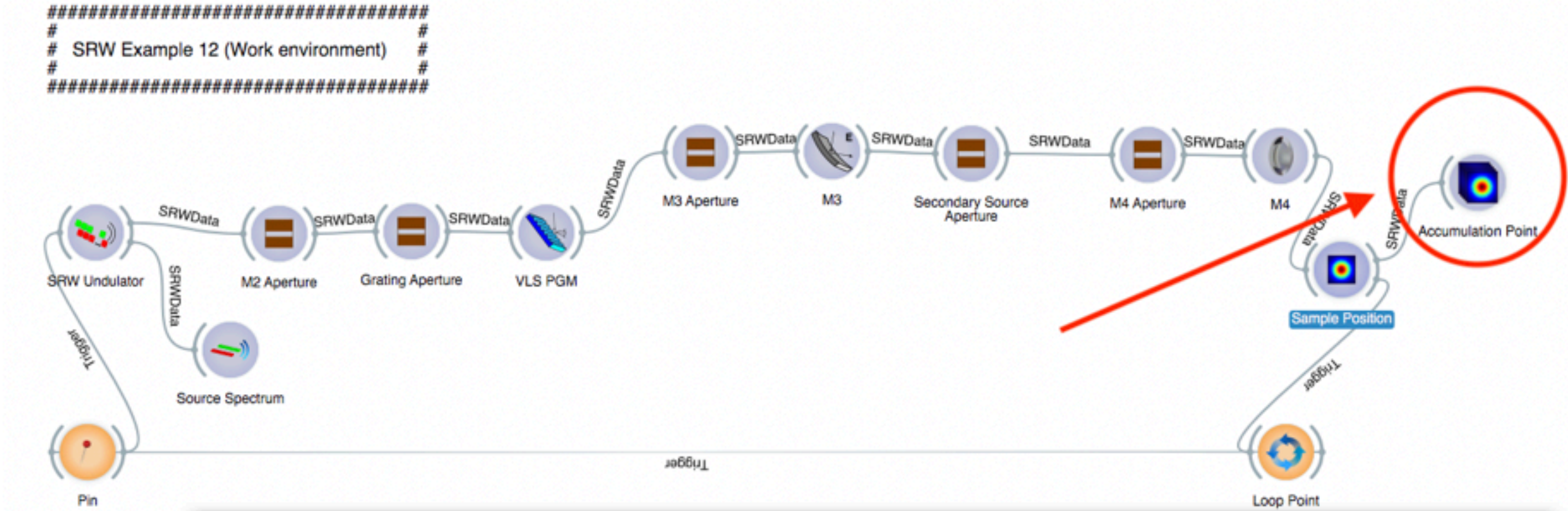




Typically, the Element by element modality is useful when designing the beamline layout and setting up the simulation parameters. The Whole Beamline modality is useful when the layout is fixed or to efficiently make loops.

MULTI-ELECTRON TOOLS

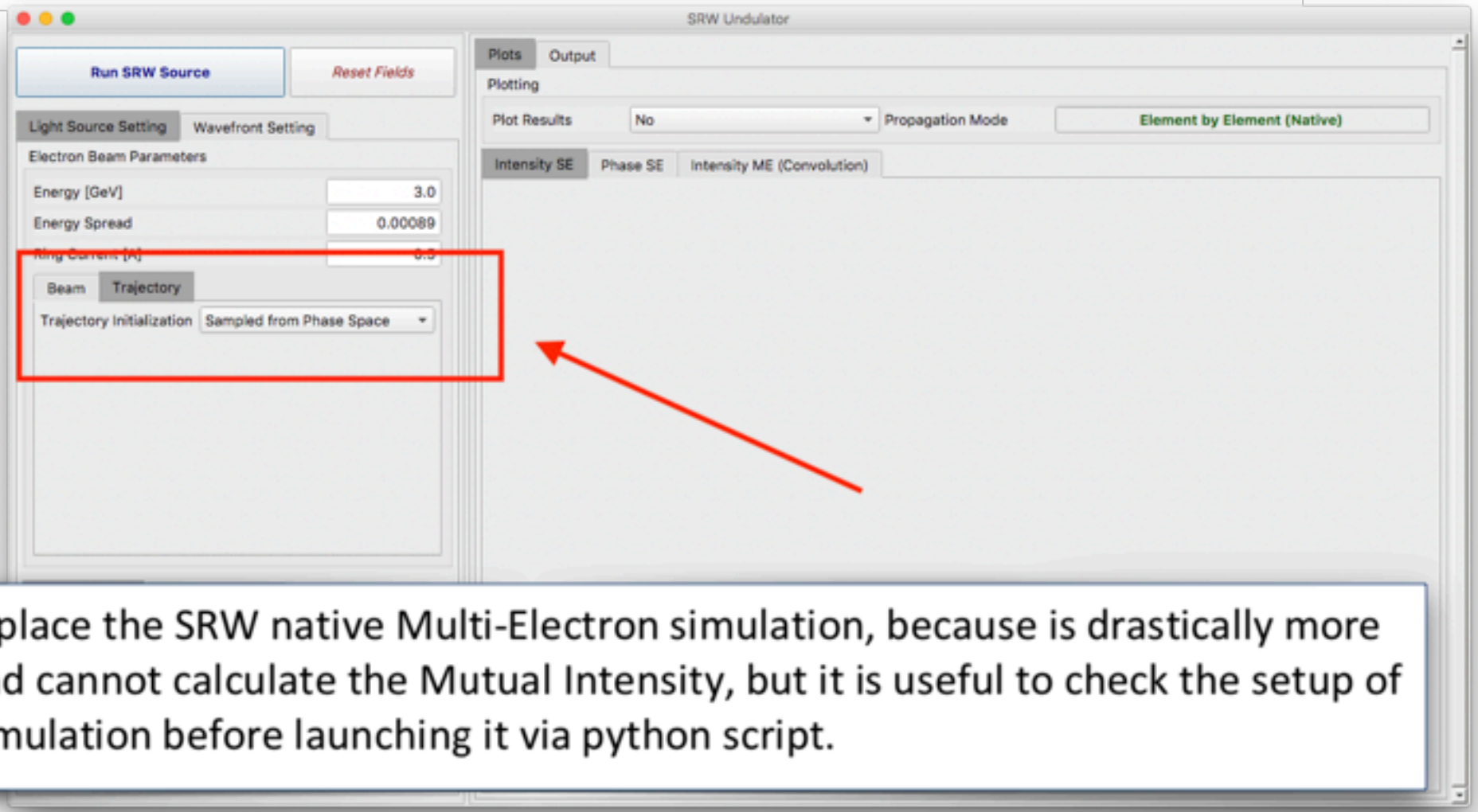
Multi-Electron Loops



Loops can provide an "online" Multi-Electron simulations (calculating intensity). The Loop Point should be connected after the "Final Screen" widget (see before), and an Accumulation Point Widget should be attached after the Final Screen, outside the loop (see Figure). The accumulation point will plot the Single-Electron intensity iteration by iteration, overlapping it to the previously calculated ones, progressively showing the resulting Multi-Electron intensity.

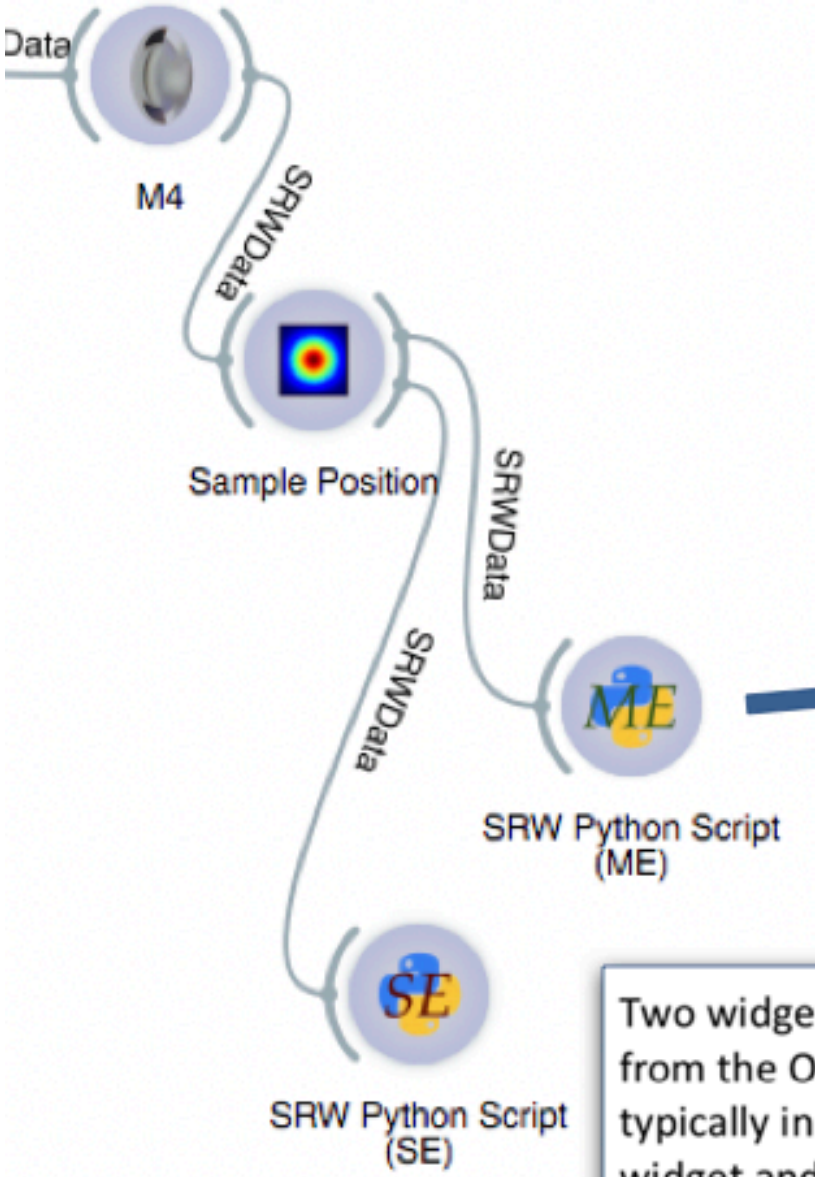
Multi-Electron Loops

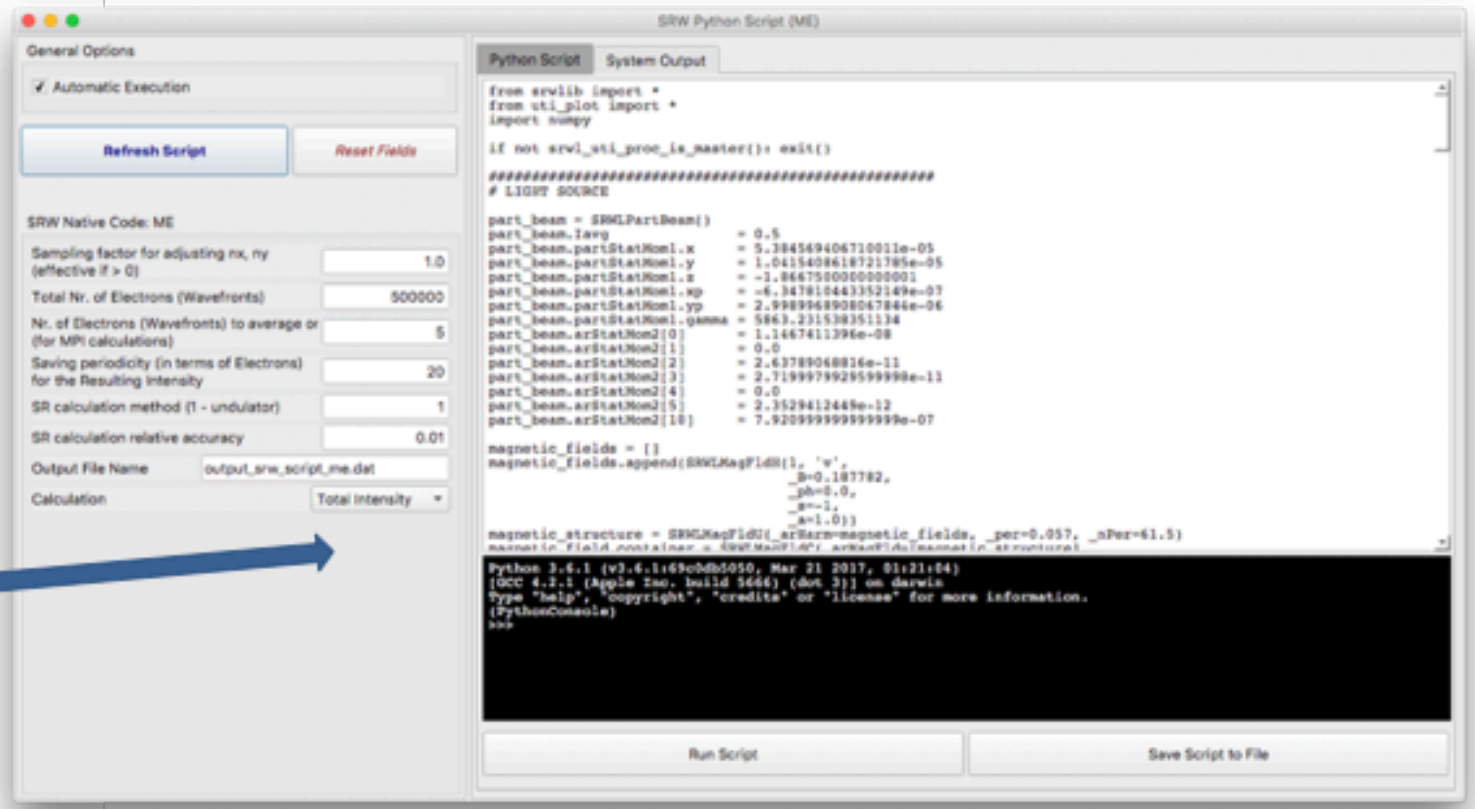
The mechanism has a meaning only when the Trajectory Initialization is set to “Sampled from Phase Space” in the Source (see Figure).



This tool cannot replace the SRW native Multi-Electron simulation, because is drastically more time consuming and cannot calculate the Mutual Intensity, but it is useful to check the setup of a Multi-Electron simulation before launching it via python script.

Native SRW Tools





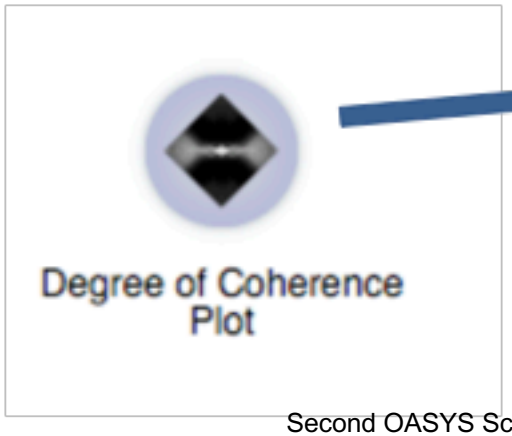
Two widgets provide a native SRW python script by automatically extracting the information from the OASYS simulation, giving the possibility to any user to run the simulation externally, typically in an optimized environment like a cluster. The widget must be connected to the final widget and, after running the simulation, the will extract and show the script. It is possible to run the script (to check them) from the widget input form.

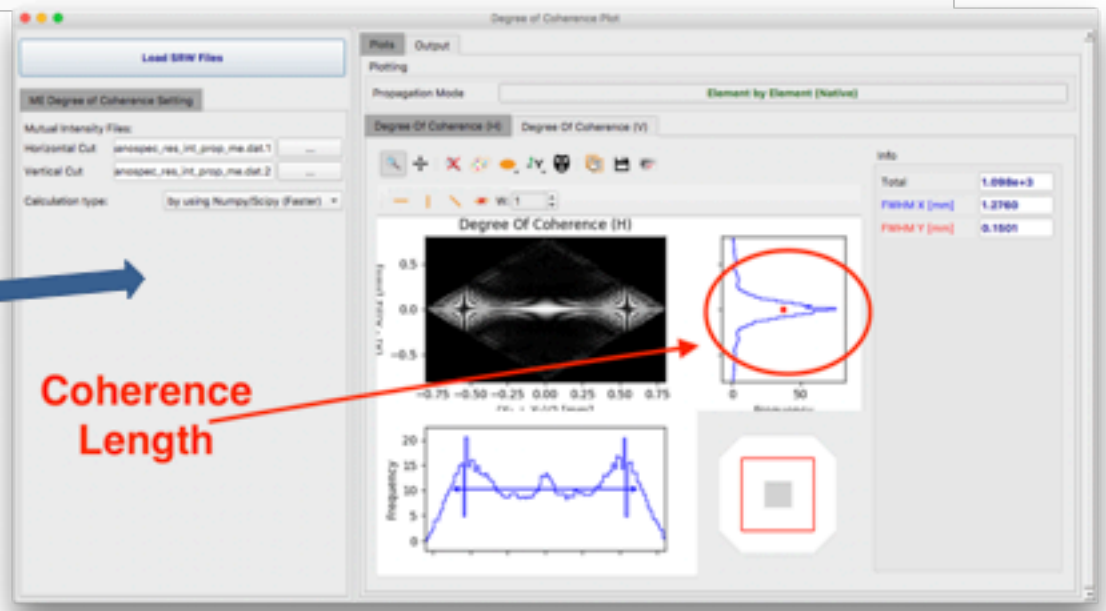
Native SRW Tools

In particular the SRW Python Script (ME) create a Multi-Electron simulation script to be saved and launched in a proper environment. Launch such a simulation within OASYS is highly discouraged.

The SRW Multi-Electron Simulation must be set up to provide or the Total Intensity or the Mutual Intensity, providing different kind of files. In the Mutual Intensity calculation, the horizontal and vertical cuts (at 0 X,Y position) are provided, too.

Two widgets help the user to correctly read and process the data:





Thank you!