Beamline components glossary

Introduction

This draft document contains a suggestion for the parameters needed to describe the beamline source and components. Its aim is to serve as an input for Optics software (targeting, for example, a future GUI for SHADOW and SRW) and for defining parameters in a HDF5 or Nexus formatted files. It could be a starting point to be able in the future to exchange information and results among different codes.

The initial idea is to reduce to the minimum the number of parameters. Each component parameters are "intrinsic", i.e., not related to the orientation, positioning or alignment in the beamline, which should be defined as external attributes.

Type of components:

- Ideal components, the components reduced to simplest idealizations (using prefix IC_)
- Base components, to define as close as possible the real ones (using prefix BC_)
- Compound components (using prefix CC_), made by combination or repetition of ideal or base components

Note that this document describes only the physical parameters of the components. The positioning and alignment attributes are yet undefined, as well as the calculation parameters (e.g., number of sampling points, grids, scans, etc.) that will depend on the kind of simulation to be done.

Glossary: BM (Bending magnet), ID (insertion device), H (horizontal), V (vertical)

IC_PhotonBeamPencil

Name	Type	Units	Description and values
energyMin	Float	eV	Minimum photon energy
energyMax	Float	eV	Maximum photon energy

IC_DriftSpace

Name	Type	Units	Description and values
d	Float	m	distance

IC_Lens

Name	Type	Units	Description and values
FH	Float	m	Focal length in H
FV	Float	m	Focal length in V

BC_ElectronBeamGaussian

Name	Type	Units	Occurrence	Description and values
ElectronEnergy	Float	GeV		Electron energy in the storage ring
ElectronCurrent	Float	A		Electron current intensity
OrbitOffset	Float(6)			Orbit offset (x,x',y,y',s,δ) from where initial conditions are defined
InputType	Integer			0=Twiss description 1=Full description
ElectronEnergySpread	Float	Adimensional	InputType=0	Spread RMS of the energy of the electrons $\Delta E/E$
EmittanceH	Float	m.rad		Horizontal emittance (at waist)
EmittanceV	Float	m.rad		Vertical emittance (at waist)
BetaH	Float	m		Beta function (horizontal)
BetaV	Float	m		Beta function (vertical)
AlphaH	Float			Alpha function of the Twiss parameter (horizontal)
AlphaV	Float			Alpha function of the Twiss parameter (vertical)
BuchLength	Float	m		Bunch length
DispersionH	Float	m		Dispersion H
DispersionV	Float	m		Dispersion V
DispersionDerivH	Float	Adimensional		Dispersion derivative H
DispersionDerivV	Float	Adimensional		Dispersion derivative V
SIGMA matrix	Float(6,6)		InputType=1	See appendix
M matrix	Float(6,6)			

BC_BendingMagnet

Name	Type	Units	Description and values
MagneticField	Float	T	Bending magnet magnetic field
MagneticFieldErrors	Float(2,N)	%	Tabulation of magnetic field errors
HorizontalArc	Float	mrad	length (angular) of the BM

BC_InsertionDevice

Name	Type	Units	Occurrence	Description and values
Type	String			Wiggler, undulator, elliptical w/u
InputType	Integer			0=Reduced description 1=Full description (B from Harmonics) 2=Full description (B from table)
PeriodID	Float	m		ID period
N	Integer			Number of periods
Kh	Float	Adimensional	InputType=0	Horizontal K value
Kv	Float	Adimensional		Vertical K value
phase	Float	rad		Phase between H and V magnets
taperH	Float	% of Kh		Gap taper H
taperV	Float	% of Kv		Gap taper V
Bharmonics	Float(N,2)	T	InputType=1	List of N harmonics, and their H and V intensities
Btable	Float(2,N)	T	InputType=2	(BHorizontal,Bvertical) vs s

BC_Slit (slit or aperture)

Name	Type	Units	Occurrence	Description and values

centerH	Float	m		H center
centerV	Float	m		V center
shape	Integer			0: None (fully opened) 1: rectangular 2: elliptical 3: free form (polygon)
Stop	Integer			0: (No) aperture 1: (yes) beam stop
gapH	Float	m	shape=1,2	H gap (twice H semiaxis for ellipse)
gapV	Float	m		V gap (twice V semiaxis for ellipse)
coordH	Float(N)	m	shape=3	H coordinates for polygon defining the shape
coordV	Float(N)	m		H coordinates for polygon defining the shape

BC_OpticalSurface

Name	Type	Units	Description and values
limits	Integer		0: Infinite surface 1: rectangular 2: elliptical 3: free form
length	Float	m	length (twice H semiaxis for ellipse)
width	Float	m	width (twice V semiaxis for ellipse)
shape	Integer(2)		0: Plane 1: Conic • 1,0: by coefficients • 1,1: sphere • 1,2: ellipsoid • 1,3: paraboloid • 1,4: hyperboloid 2: Toroid 3: Free • 3,1: Mesh • 3,2: Polynomial
coeff	Float(10)		Coefficients, radii, semiaxes, etc. depending on shape
Geometry	Integer		0: reflecting (e.g., mirrors) 1: transmitting (e.g., lenses, Laue crystals) 2: both (e.g., diamond crystals, beamsplitters)

BC_Attenuator (attenuator or filter)

Name	Type	Units	Description and values
material	string		String describing the material (e.g., Cu, H2O, etc).
thickness	Float	m	Attenuator thickness
density	Float	g/cm^3	Material density

BC_Mirror

Name	Type	Units	Description and values
coating	string		String describing the material (e.g., Cu, H2O, etc).
thickness	Float	m	coating thickness
density	Float	g/cm^3	coating density

BC_CrystalPerfect

Name	Type	Units	Description and values
name	string		String describing the material (e.g., Si, quartz, etc).
thickness	Float	m	crystal thickness
Cell	Float(6)	a,b,c (Angstroms) alpha, beta, gamma (deg)	Crystallographic cell parameters
N	Integer		Number of atoms in unit cell
Z	Integer(N)		Atomic number of atoms in unit cell
X,Y,Z	Float(N)	Angstroms	Coordinates of atoms in crystallographic cell
occupancy	Float(N)		Occupancy coeff of atoms in unit cell
Temperature0	Float	K	Temperature at which unit cell is given
Temperature	Float	K	Crystal temperature
Miller	Integer(3)		Miller indices of selected reflection
Asymmetry angle	Float	Deg	Asymmetry angle (better define in vector form?)

BC_Multilayer

Name	Type	Units	Occurrence	Description and values

aperiodic	Integer			0: (No) Periodic multilayer 1: (Yes) Free multilayer
LateralGradient	Integer			0: No 1: Yes (linear variation) 2: Yes (Gamma2 vs length coordinate)
LateralGradientCoeff	Float(2)		LateralGradiant=1	Coefficients for: Gamma2 = LateralGradientCoeff[0] + LateralGradientCoeff[1]*length_coordinate
LateralGradientArray	Float (2,npts)		LateralGradiant=2	Gamma2 vs length_coordinate
MaterialSubstrate	string			String describing the material (e.g., Si, W, B4C)
DensitySubstrate	Float	g/cm^3		Density of substrate
RoughnessSubstrate	Float	Angstroms		Roughness RMS of substrate
MaterialSublayer1	string		aperiodic=0	String describing the material (e.g., Si)
MaterialSublayer2	string			String describing the material (e.g., Si)
MaterialToplayer	string			String describing the material (e.g., Si)
Period	Float	Angstroms		Period of main layers (thickness of sublayer1 plus thickness of sublayer2)
Gamma2	Float			Ratio: thickness_sublayer2/Period
DensitySublayer1	Float	g/cm^3		Density of sublayer1
DensitySublayer2	Float	g/cm^3		Density of sublayer2
DensityToplayer1	Float	g/cm^3		Density of toplayer
RoughnessSublayer1	Float	Angstroms		Roughness RMS of sublayer1
RoughnessSublayer2	Float	Angstroms		Roughness RMS of sublayer2
RoughnessToplayer	Float	Angstroms		Roughness RMS of toplayer
ThicknessToplater	Float	Angstroms		thickness of toplayer (ero means no toplayer)
MaterialArray	String(N)		aperiodic=1	Material for individual layers
ThicknessArray	Float	Angstroms		Thickness of individual layers
DensityArray	Float(N)	g/cm^3		Density of individual layers
RoughnessArray	Float	Angstroms		Roughness RMS of individual layers

BC_LensSingle

Name	Type	Units	Description and values

material	string		String describing the material (e.g., Si, Be, etc).
thickness	Float	m	lens thickness
density	Float	g/cm^3	lens density

Compound elements

Name	Needs	Description and values
CC_Source	BC_ElectronBeamGaussian and BC_bendingMagnet or BC_InsertionDevice	Synchrotron source
CC_Monochromator	BC_CrystalPerfect(N) or/and BC_Multilayer	monochromator
CC_CRL	BC_LensSingle(N)	CRL as an array of single lenses
CC_Transfocator	CC_CRL(N)	Transfocator as an array of CRLs

To do list, some thoughts and questions

Add ideal and basic components:

- IC_PhotonBeamGaussian
- IC_PhotonBeamGeometric (geometrical source, like in SHADOW)
- BC_Grating
- BC_CapillarySingle
- BC_CapillaryMulti (or BC_Kumakhov)

How to represent undefined parameters

- Skip mentioning them in the list
- Assign a default value (this may be confused)
- Assign a "None" value (like in python)

Discuss on how to position components:

- Distance to previous component (using IC_DriftSpace), or distance to previous element (along optical path)
- Define full orientation by either
    - Default
    - Incident angle (tangential), orientation angle (sagittal)

    - Incident angle, "mirror orientation angle", and "mirror movement (6)" (as in SHADOW)
    - Pitch, roll and yaw angles?
    - Euler angles?
    - Other?
- Define automatic positioning as a function of the "working conditions" (e.g., define working photon energy, and calculate Bragg angles?)
- How to define and position the monitors, observation planes, "detectors", etc. Should be defines as "components" (like in McXtrace)?

Discuss in the name of variables:

- Capitalized? Using underscores? Using capital letters to separate words?
- Long versus short (abbreviated) names?

Units:

- Shall we adopt as far as possible the SI or cgs? (e.g. , all lengths in m, temperatures in K, etc)
- Shall we prefer not to use multipliers (e.g., eV preferred to GeV or keV, etc)
- Shall we provide (as in Nexus) the possibility of using different units?

Names:

- Shall we add the possibility to add parameters depending on the user needs? If so, shall we define a "used defined" prefix?
- Shall we add the possibility to define "derived" parameters, e.g., wavelength as a function of the energy (WAVELENGTH=12.39842/ENERGY)?

Could basic components use other basic components? (e.g., I defined the optical surface shape separated from the nature of the element (mirror, crystal, etc). Or only "compound" components could do that?