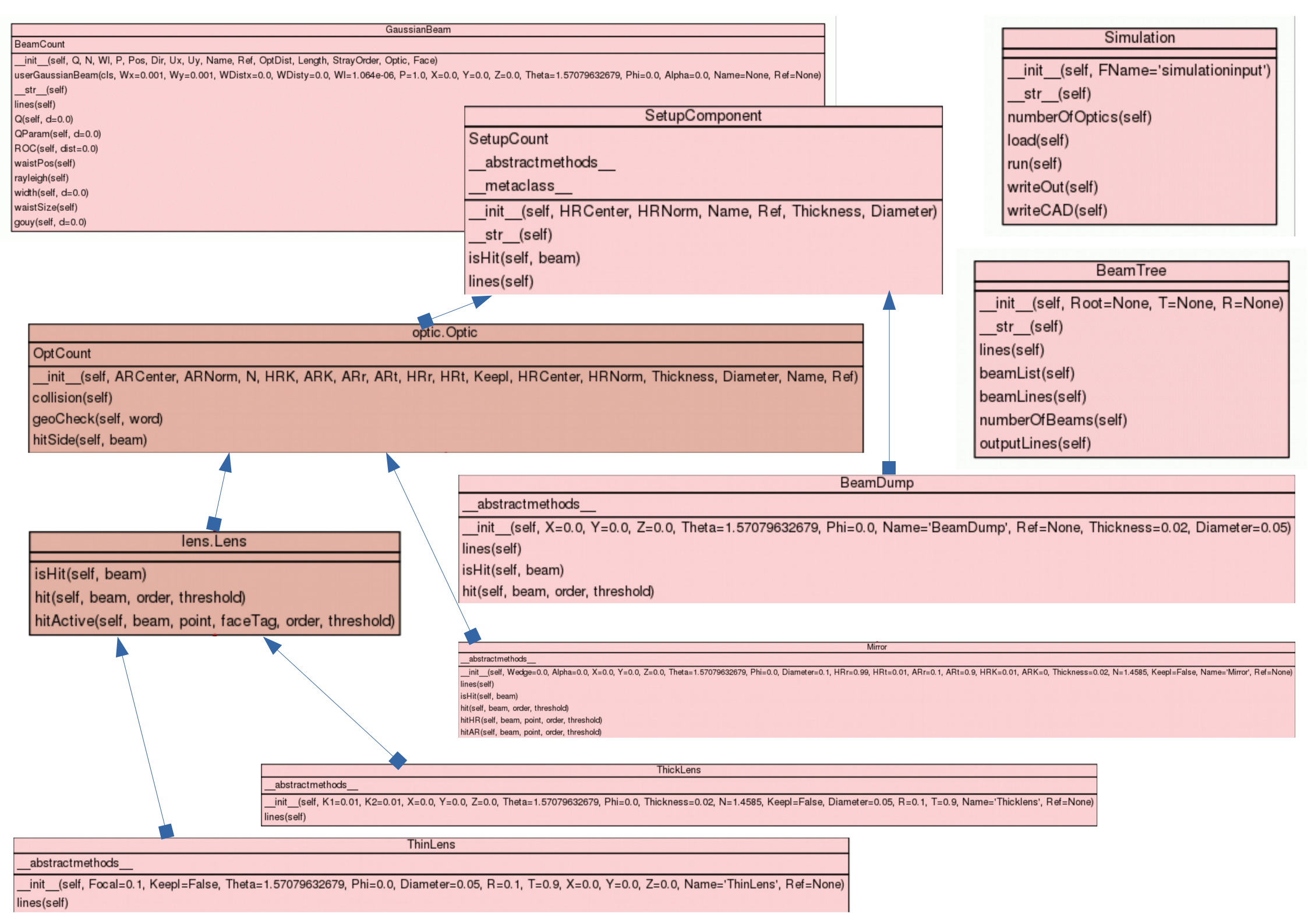
GaussianBeam
BeamCount
__init__(self, Q, N, Wl, P, Pos, Dir, Ux, Uy, Name, Ref, OptDist, Length, StrayOrder, Optic, Face)
userGaussianBeam(cls, Wx=0.001, Wy=0.001, WDistx=0.0, WDisty=0.0, Wl=1.064e-06, P=1.0, X=0.0, Y=0.0, Z=0.0, Theta=1.57079632679, Phi=0.0, Alpha=0.0, Name=None, Ref=None)
__str__(self)
lines(self)
Q(self, d=0.0)
QParam(self, d=0.0)
ROC(self, dist=0.0)
waistPos(self)
rayleigh(self)
width(self, d=0.0)
waistSize(self)
gouy(self, d=0.0)
SetupComponent
SetupCount
__abstractmethods__
__metaclass__
__init__(self, HRCenter, HRNorm, Name, Ref, Thickness, Diameter)
__str__(self)
isHit(self, beam)
lines(self)
Simulation
__init__(self, FName='simulationinput')
__str__(self)
numberOfOptics(self)
load(self)
run(self)
writeOut(self)
writeCAD(self)
BeamTree
__init__(self, Root=None, T=None, R=None)
__str__(self)
lines(self)
beamList(self)
beamLines(self)
numberOfBeams(self)
outputLines(self)
optic.Optic
OptCount
__init__(self, ARCenter, ARNorm, N, HRK, ARK, ARr, ARt, HRr, HRt, Keepl, HRCenter, HRNorm, Thickness, Diameter, Name, Ref)
collision(self)
geoCheck(self, word)
hitSide(self, beam)
BeamDump
__abstractmethods__
__init__(self, X=0.0, Y=0.0, Z=0.0, Theta=1.57079632679, Phi=0.0, Name='BeamDump', Ref=None, Thickness=0.02, Diameter=0.05)
lines(self)
isHit(self, beam)
hit(self, beam, order, threshold)
lens.Lens
isHit(self, beam)
hit(self, beam, order, threshold)
hitActive(self, beam, point, faceTag, order, threshold)
Mirror
__abstractmethods__
__init__(self, Wedge=0.0, Alpha=0.0, X=0.0, Y=0.0, Z=0.0, Theta=1.57079632679, Phi=0.0, Diameter=0.1, HRr=0.99, HRt=0.01, ARr=0.1, ARt=0.9, HRK=0.01, ARK=0, Thickness=0.02, N=1.4585, Keepl=False, Name='Mirror', Ref=None)
lines(self)
isHit(self, beam)
hit(self, beam, order, threshold)
hitHR(self, beam, point, order, threshold)
hitAR(self, beam, point, order, threshold)
ThickLens
__abstractmethods__
__init__(self, K1=0.01, K2=0.01, X=0.0, Y=0.0, Z=0.0, Theta=1.57079632679, Phi=0.0, Thickness=0.02, N=1.4585, Keepl=False, Diameter=0.05, R=0.1, T=0.9, Name='Thicklens', Ref=None)
lines(self)
ThinLens
__abstractmethods__
__init__(self, Focal=0.1, Keepl=False, Theta=1.57079632679, Phi=0.0, Diameter=0.05, R=0.1, T=0.9, X=0.0, Y=0.0, Z=0.0, Name='ThinLens', Ref=None)
lines(self)