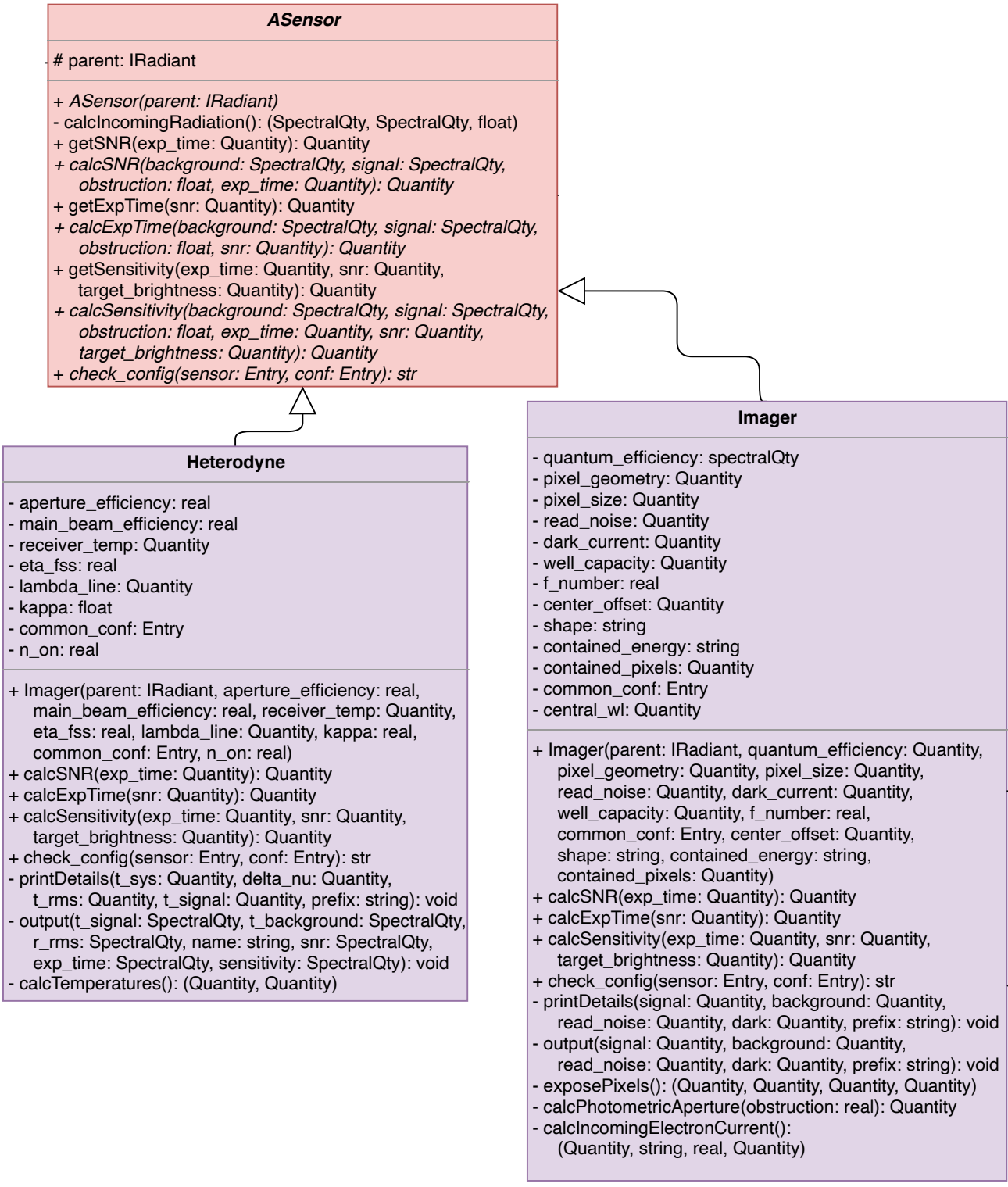
ASensor
parent: IRadiant
+ ASensor(parent: IRadiant)
- calcIncomingRadiation(): (SpectralQty, SpectralQty, float)
+ getSNR(exp_time: Quantity): Quantity
+ calcSNR(background: SpectralQty, signal: SpectralQty, obstruction: float, exp_time: Quantity): Quantity
+ getExpTime(snr: Quantity): Quantity
+ calcExpTime(background: SpectralQty, signal: SpectralQty, obstruction: float, snr: Quantity): Quantity
+ getSensitivity(exp_time: Quantity, snr: Quantity, target_brightness: Quantity): Quantity
+ calcSensitivity(background: SpectralQty, signal: SpectralQty, obstruction: float, exp_time: Quantity, snr: Quantity, target_brightness: Quantity): Quantity
+ check_config(sensor: Entry, conf: Entry): str
Heterodyne
- aperture_efficiency: real
- main_beam_efficiency: real
- receiver_temp: Quantity
- eta_fss: real
- lambda_line: Quantity
- kappa: float
- common_conf: Entry
- n_on: real
+ Imager(parent: IRadiant, aperture_efficiency: real, main_beam_efficiency: real, receiver_temp: Quantity, eta_fss: real, lambda_line: Quantity, kappa: real, common_conf: Entry, n_on: real)
+ calcSNR(exp_time: Quantity): Quantity
+ calcExpTime(snr: Quantity): Quantity
+ calcSensitivity(exp_time: Quantity, snr: Quantity, target_brightness: Quantity): Quantity
+ check_config(sensor: Entry, conf: Entry): str
- printDetails(t_sys: Quantity, delta_nu: Quantity, t_rms: Quantity, t_signal: Quantity, prefix: string): void
- output(t_signal: SpectralQty, t_background: SpectralQty, r_rms: SpectralQty, name: string, snr: SpectralQty, exp_time: SpectralQty, sensitivity: SpectralQty): void
- calcTemperatures(): (Quantity, Quantity)
Imager
- quantum_efficiency: spectralQty
- pixel_geometry: Quantity
- pixel_size: Quantity
- read_noise: Quantity
- dark_current: Quantity
- well_capacity: Quantity
- f_number: real
- center_offset: Quantity
- shape: string
- contained_energy: string
- contained_pixels: Quantity
- common_conf: Entry
- central_wl: Quantity
+ Imager(parent: IRadiant, quantum_efficiency: Quantity, pixel_geometry: Quantity, pixel_size: Quantity, read_noise: Quantity, dark_current: Quantity, well_capacity: Quantity, f_number: real, common_conf: Entry, center_offset: Quantity, shape: string, contained_energy: string, contained_pixels: Quantity)
+ calcSNR(exp_time: Quantity): Quantity
+ calcExpTime(snr: Quantity): Quantity
+ calcSensitivity(exp_time: Quantity, snr: Quantity, target_brightness: Quantity): Quantity
+ check_config(sensor: Entry, conf: Entry): str
- printDetails(signal: Quantity, background: Quantity, read_noise: Quantity, dark: Quantity, prefix: string): void
- output(signal: Quantity, background: Quantity, read_noise: Quantity, dark: Quantity, prefix: string): void
- exposePixels(): (Quantity, Quantity, Quantity, Quantity)
- calcPhotometricAperture(obstruction: real): Quantity
- calcIncomingElectronCurrent(): (Quantity, string, real, Quantity)