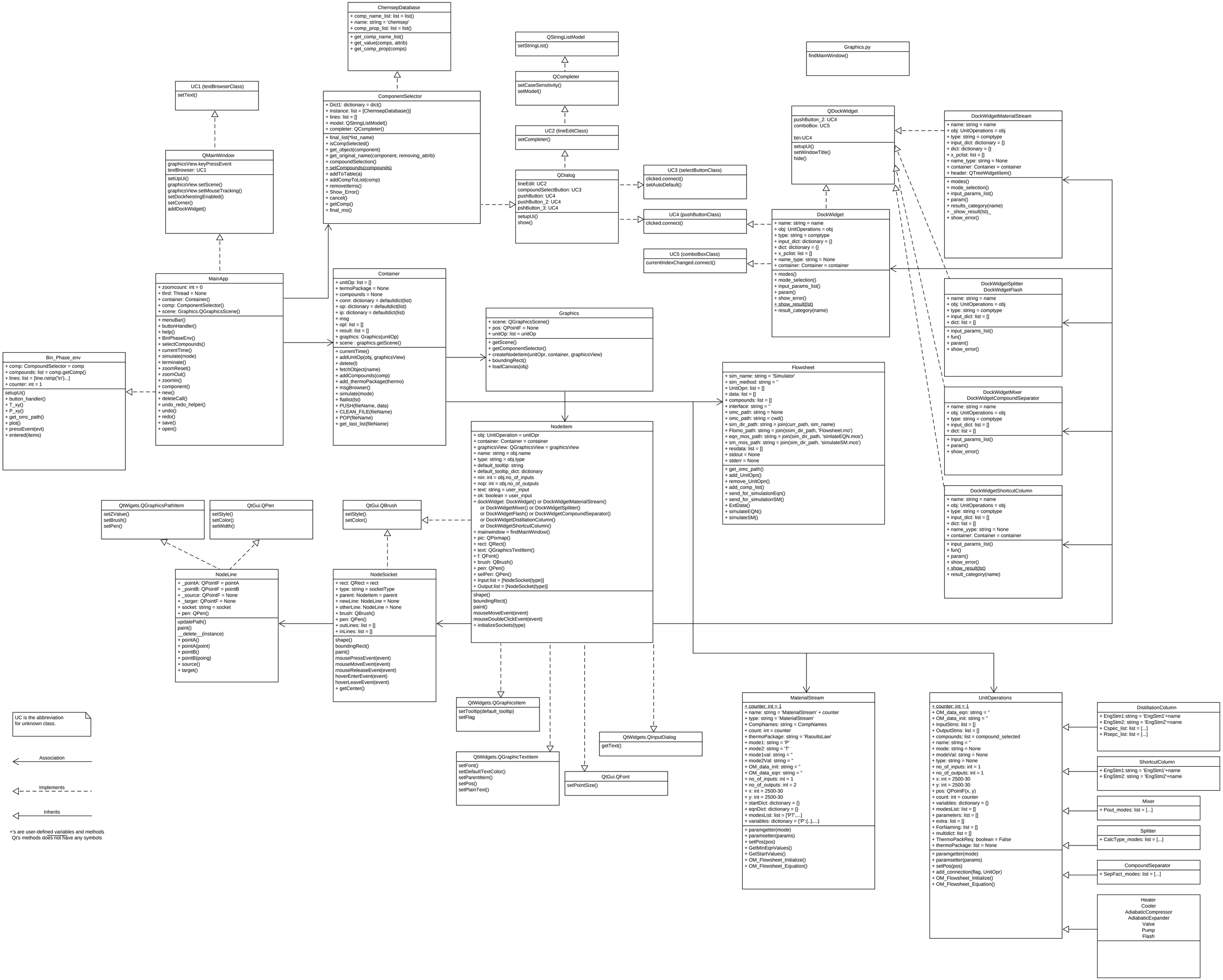
ChemsepDatabase
+ comp_name_list: list = list()
+ name: string = 'chemsep'
+ comp_prop_list: list = list()
+ get_comp_name_list()
+ get_value(comps, attrib)
+ get_comp_prop(comps)
QStringListModel
setStringList()
QCompleter
setCaseSensitivity()
setModel()
UC2 (lineEditClass)
setCompleter()
Graphics.py
findMainWindow()
UC1 (textBrowserClass)
setText()
ComponentSelector
+ Dict1: dictionary = dict()
+ instance: list = [ChemsepDatabase()]
+ lines: list = []
+ model: QStringListModel()
+ completer: QCompleter()
+ final_list(*list_name)
+ isCompSelected()
+ get_object(component)
+ get_original_name(component, removing_attrib)
+ compoundSelection()
+ setCompounds(compounds)
+ addToTable(a)
+ addCompToList(comp)
+ removeItems()
+ Show_Error()
+ cancel()
+ getComp()
+ final_mo()
QDialog
lineEdit: UC2
compoundSelectButton: UC3
pushButton: UC4
pushButton_2: UC4
pshButton_3: UC4
setupUi()
show()
UC3 (selectButtonClass)
clicked.connect()
setAutoDefault()
UC4 (pushButtonClass)
clicked.connect()
UC5 (comboBoxClass)
currentIndexChanged.connect()
QDockWidget
pushButton_2: UC4
comboBox: UC5
btn:UC4
setupUi()
setWindowTitle()
hide()
DockWidgetMaterialStream
+ name: string = name
+ obj: UnitOperations = obj
+ type: string = comptype
+ input_dict: dictionary = {}
+ dict: dictionary = {}
+ x_pclist: list = []
+ name_type: string = None
+ container: Container = container
+ header: QTreeWidgetItem()
+ modes()
+ mode_selection()
+ input_params_list()
+ param()
+ results_category(name)
+ _show_result(lst)_
+ show_error()
QMainWindow
graphicsView.keyPressEvent
textBrowser: UC1
setUpUi()
graphicsView.setScene()
graphicsView.setMouseTracking()
setDockNestingEnabled()
setCorner()
addDockWidget()
DockWidget
+ name: string = name
+ obj: UnitOperations = obj
+ type: string = comptype
+ input_dict: dictionary = {}
+ dict: dictionary = {}
+ x_pclist: list = []
+ name_type: string = None
+ container: Container = container
+ modes()
+ mode_selection()
+ input_params_list()
+ param()
+ show_error()
+ show_result(lst)
+ result_category(name)
MainApp
+ zoomcount: int = 0
+ thrd: Thread = None
+ container: Container()
+ comp: ComponentSelector()
+ scene: Graphics.QGraphicsScene()
+ menuBar()
+ buttonHandler()
+ help()
+ BinPhaseEnv()
+ selectCompounds()
+ currentTime()
+ simulate(mode)
+ terminate()
+ zoomReset()
+ zoomOut()
+ zoomIn()
+ component()
+ new()
+ deleteCall()
+ undo_redo_helper()
+ undo()
+ redo()
+ save()
+ open()
Container
+ unitOp: list = []
+ termoPackage = None
+ compounds = None
+ conn: dictionary = defaultdict(list)
+ op: dictionary = defaultdict(list)
+ ip: dictionary = defaultdict(list)
+ msg
+ opl: list = []
+ result: list = []
+ graphics: Graphics(unitOp)
+ scene : graphics.getScene()
+ currentTime()
+ addUnitOp(obj, graphicsView)
+ delete(I)
+ fetchObject(name)
+ addCompounds(comp)
+ add_thermoPackage(thermo)
+ msgBrowser()
+ simulate(mode)
+ flatlist(lst)
+ PUSH(fileName, data)
+ CLEAN_FILE(fileName)
+ POP(fileName)
+ get_last_list(fileName)
Graphics
+ scene: QGraphicsScene()
+ pos: QPointF = None
+ unitOp: list = unitOp
+ getScene()
+ getComponentSelector()
+ createNodeItem(unitOpr, container, graphicsView)
+ boundingRect()
+ loadCanvas(obj)
DockWidgetSplitter
DockWidgetFlash
+ name: string = name
+ obj: UnitOperations = obj
+ type: string = comptype
+ input_dict: list = []
+ dict: list = []
+ input_params_list()
+ fun()
+ param()
+ show_error()
Bin_Phase_env
+ comp: CompoundSelector = comp
+ compounds: list = comp.getComp()
+ lines: list = [line.rstrip('\n')...]
+ counter: int = 1
setupUi()
+ button_handler()
+ T_xy()
+ P_xy()
+ get_omc_path()
+ plot()
+ pressEvent(evt)
+ entered(items)
Flowsheet
+ sim_name: string = 'Simulator'
+ sim_method: string = ''
+ UnitOpn: list = []
+ data: list = []
+ compounds: list = []
+ interface: string = ''
+ omc_path: string = None
+ omc_path: string = cwd()
+ sim_dir_path: string = join(curr_path, sim_name)
+ Flomo_path: string = join(ssim_dir_path, 'Flowsheet.mo')
+ eqn_mos_path: string = join(sim_dir_path, 'simlateEQN.mos')
+ sm_mos_path: string = join(sim_dir_path, 'simulateSM.mos')
+ resdata: list = []
+ stdout = None
+ stderr = None
+ get_omc_path()
+ add_UnitOpn()
+ remove_UnitOpn()
+ add_comp_list()
+ send_for_simulationEqn()
+ send_for_simulationSM()
+ ExtData()
+ simulateEQN()
+ simulateSM()
DockWidgetMixer
DockWidgetCompoundSeparator
+ name: string = name
+ obj: UnitOperations = obj
+ type: string = comptype
+ input_dict: list = []
+ dict: list = []
+ input_params_list()
+ param()
+ show_error()
NodeItem
+ obj: UnitOperation = unitOpr
+ container: Container = container
+ graphicsView: QGraphicsView = graphicsView
+ name: string = obj.name
+ type: string = obj.type
+ default_tooltip: string
+ default_tooltip_dict: dictionary
+ nin: int = obj.no_of_inputs
+ nop: int = obj.no_of_outputs
+ text: string = user_input
+ ok: boolean = user_input
+ dockWidget: DockWidget() or DockWidgetMaterialStream()
or DockWidgetMixer() or DockWidgetSplitter()
or DockWidgetFlash() or DockWidgetCompoundSeparator()
or DockWidgetDistillationColumn()
or DockWidgetShortcutColumn()
+ mainwindow = findMainWindow()
+ pic: QPixmap()
+ rect: QRect()
+ text: QGraphicsTextItem()
+ f: QFont()
+ brush: QBrush()
+ pen: QPen()
+ selPen: QPen()
+ Input:list = [NodeSocket(type)]
+ Output:list = [NodeSocket(type)]
shape()
boundingRect()
paint()
mouseMoveEvent(event)
mouseDoubleClickEvent(event)
+ initializeSockets(type)
DockWidgetShortcutColumn
+ name: string = name
+ obj: UnitOperations = obj
+ type: string = comptype
+ input_dict: list = []
+ dict: list = []
+ name_ype: string = None
+ container: Container = container
+ input_params_list()
+ fun()
+ param()
+ show_error()
+ show_result(lst)
+ result_category(name)
QtWigets.QGraphicsPathItem
setZValue()
setBrush()
setPen()
QtGui.QPen
setStyle()
setColor()
setWidth()
QtGui.QBrush
setStyle()
setColor()
NodeLine
+ _pointA: QPointF = pointA
+ _pointB: QPointF = pointB
+ _source: QPointF = None
+ _targer: QPointF = None
+ socket: string = socket
+ pen: QPen()
updatePath()
paint()
__delete__(instance)
+ pointA()
+ pointA(point)
+ pointB()
+ pointB(poing)
+ source()
+ target()
NodeSocket
+ rect: QRect = rect
+ type: string = socketType
+ parent: NodeItem = parent
+ newLine: NodeLine = None
+ otherLine: NodeLine = None
+ brush: QBrush()
+ pen: QPen()
+ outLines: list = []
+ inLines: list = []
shape()
boundingRect()
paint()
mousePressEvent(event)
mouseMoveEvent(event)
mouseReleaseEvent(event)
hoverEnterEvent(event)
hoverLeaveEvent(event)
+ getCenter()
QtWidgets.QGraphicsItem
setTooltip(default_tooltip)
setFlag
QtWidgets.QInputDialog
getText()
QtWidgets.QGraphicTextItem
setFont()
setDefaultTextColor()
setParentItem()
setPos()
setPlainText()
QtGui.QFont
setPointSize()
MaterialStream
+ counter: int = 1
+ name: string = 'MaterialStream' + counter
+ type: string = 'MaterialStream'
+ CompNames: string = CompNames
+ count: int = counter
+ thermoPackage: string = 'RaoultsLaw'
+ mode1: string = 'P'
+ mode2: string = 'T'
+ mode1val: string = ''
+ mode2Val: string = ''
+ OM_data_init: string = ''
+ OM_data_eqn: string = ''
+ no_of_inputs: int = 1
+ no_of_outputs: int = 2
+ x: int = 2500-30
+ y: int = 2500-30
+ startDict: dictionary = {}
+ eqnDict: dictionary = {}
+ modesList: list = ['PT',...]
+ variables: dictionary = {'P':{..},...}
+ paramgetter(mode)
+ paramsetter(params)
+ setPos(pos)
+ GetMinEqnValues()
+ GetStartValues()
+ OM_Flowsheet_Initialize()
+ OM_Flowsheet_Equation()
UnitOperations
+ counter: int = 1
+ OM_data_eqn: string = ''
+ OM_data_init: string = ''
+ InputStms: list = []
+ OutputStms: list = []
+ compounds: list = compound_selected
+ name: string = ''
+ mode: string = None
+ modeVal: string = None
+ type: string = None
+ no_of_inputs: int = 1
+ no_of_outputs: int = 1
+ x: int = 2500-30
+ y: int = 2500-30
+ pos: QPointF(x, y)
+ count: int = counter
+ variables: dictionary = {}
+ modesList: list = []
+ parameters: list = []
+ extra: list = []
+ ForNaming: list = []
+ multidict: list = []
+ ThermoPackReq: boolean = False
+ thermoPackage: list = None
+ paramgetter(mode)
+ paramsetter(params)
+ setPos(pos)
+ add_connection(flag, UnitOpr)
+ OM_Flowsheet_Initialize()
+ OM_Flowsheet_Equation()
DistillationColumn
+ EngStm1:string = 'EngStm1'+name
+ EngStm2: string = 'EngStm2'+name
+ Cspec_list: list = [...]
+ Rsepc_list: list = [...]
ShortcutColumn
+ EngStm1:string = 'EngStm1'+name
+ EngStm2: string = 'EngStm2'+name
Mixer
+ Pout_modes: list = [...]
Splitter
+ CalcType_modes: list = [...]
CompoundSeparator
+ SepFact_modes: list = [...]
Heater
Cooler
AdiabaticCompressor
AdiabaticExpander
Valve
Pump
Flash
UC is the abbreviation for unknown class.
Association
Implements
Inherits
+'s are user-defined variables and methods
Qt's methods does not have any symbols