Each Location contains a set of cell phones being serviced

Location

Double lat

Double lng

Int number

Set<CellPhone> cellPhonesServiced

CellPhone

Double lat

Double lng

Double power

Int mainPri

Int subPri

Int[] dailyConstraint

Int weeklyConstraint

Int weeklyRequired

Int dailyRequired

Int value

String name

addTime()

reduceTime()

setArea()

isOverlapping()

ScheduleSolver keeps all cellTowers in array and generates domain from information

CellTower has list of possible locations

All cellPhones are held in a set to keep track of the phones and how much constraint has been met

Each location is held in the domain which is used to pick optimal values

The actual schedule is an array of time blocks

TimeBlock contains a set of cellphones getting coverage for that block of time

TimeBlock contains a location for each cellTower

TimeBlock

Location[] towerLocations

Set<CellPhone> currentCoverage

Double maxPotential

Int day

Set<CellPhone> preference

Boolean manualOverride

addIncreasedPriority()

userOverride()

ScheduleSolver

Set<CellPhone> allCellPhones

CellTower[] cellTowers

TimeBlock[] schedule

List<List<Location>> domain

PriorityQueue<ScheduledConstraint> prioritizedScheduledConstraint

Boolean dailyFirst

setupProblem()

generateSchedule()

addScheduledConstraint()

manualOverride()

CellTower

List<Location> locations

double power

Double speed

Double[][] travelTime

String type