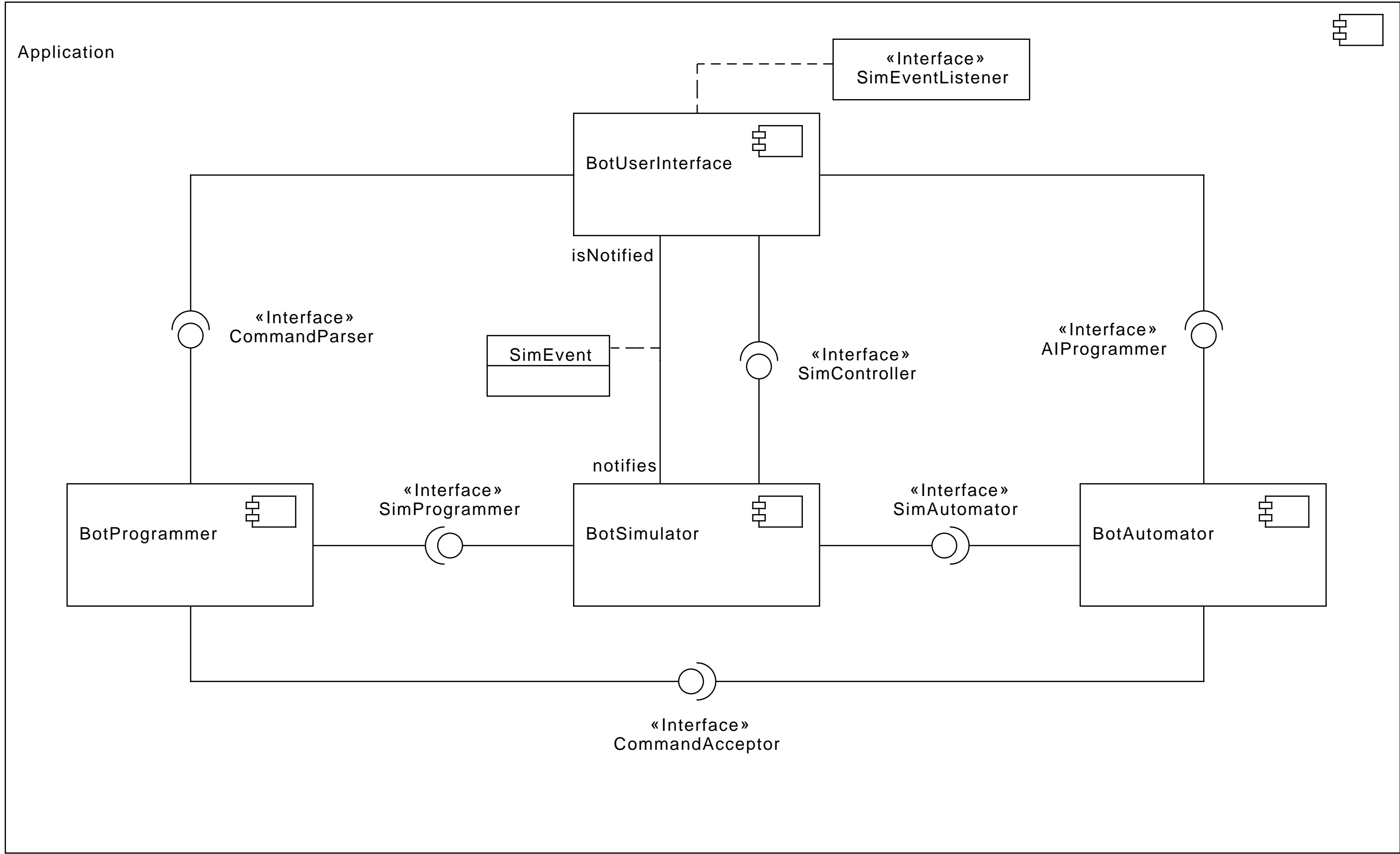


«Interface» SimEventListener
+ notifyPathAdded(PathAddedSimEvent event) + notifyPathRemoved(PathRemovedSimEvent event) + notifyRobotAdded(RobotAddedSimEvent event) + notifyRobotRemoved(RobotRemovedSimEvent event) + notifyItemAdded(ItemAddedSimEvent event) + notifyItemRemoved(ItemRemovedSimEvent event) + notifyRobotMoved(RobotMovedSimEvent event) + notifyRobotWaited(RobotWaitedSimEvent event) + notifyRobotProgrammed(RobotProgrammedSimEvent event) + notifyGridSizeChanged(GridSizeChangedSimEvent event) + notifySimStateChanged(SimStateChangedSimEvent event) + notifyItemPickedFromPath(ItemPickedFromPathSimEvent event) + notifyItemPickedFromContainer(ItemPickedFromContainerSimEvent event) + notifyItemDroppedInContainer(ItemDroppedInConatinerSimEvent event) + notifyRobotTurned(RobotTurnedSimEvent event) + notifyRobotOutOfBounds(RobotOutOfBoundsSimEvent event) + notifyRobotCollision(RobotCollisionSimEvent event) + notifyApplesInBins(ApplesInBinsSimEvent event) + notifySimStepChanged(SimStepChangedSimEvent event) + notifySimReset()



«Interface» SimController
+ setGridSize(sizeX: int, sizeY: int, sizeZ: int) + addPath(posX: int, posY: int, pos: int) + removePath(posX: int, posY: int, pos: int) + addRobot(botlD: String, botTyp: RobotType, posX: int, posY: int, posZ: int) + removeRebot(botlD: String) + addlItem(itmID: String, itmTyp: ItemType, posX: int, posY: int, posZ: int) + removeItem(itmID: String) + queryLocationState(sizeX: int, sizeY: int, sizeZ: int): LocationState + runSimForward() + runSimBackward() + stopSim() + resetSim() + stepSimForward() + stepSimBackward() + speedUpSim() + slowDownSim() + clearSim()

«Interface» SimProgrammer
+ programRobot(botlD: String, botCmdType: RobotCommandType[*])

«Interface» SimAutomator
+ getAllPaths(): Path[*] + getAllRobots(): Robot[*] + getAllItems(): Item[*]

«Interface» CommandParser
+ parseCommand(robotID: String, command: String) + setLanguage(language: LanguageType) + getProgram(robotID: String, program: String)

«Interface» CommandAcceptor
+ acceptCommands(robotlD: String, commands: RobotCommandType[*])

«Interface» AIProgrammer
+ generateAutomatedCommands()