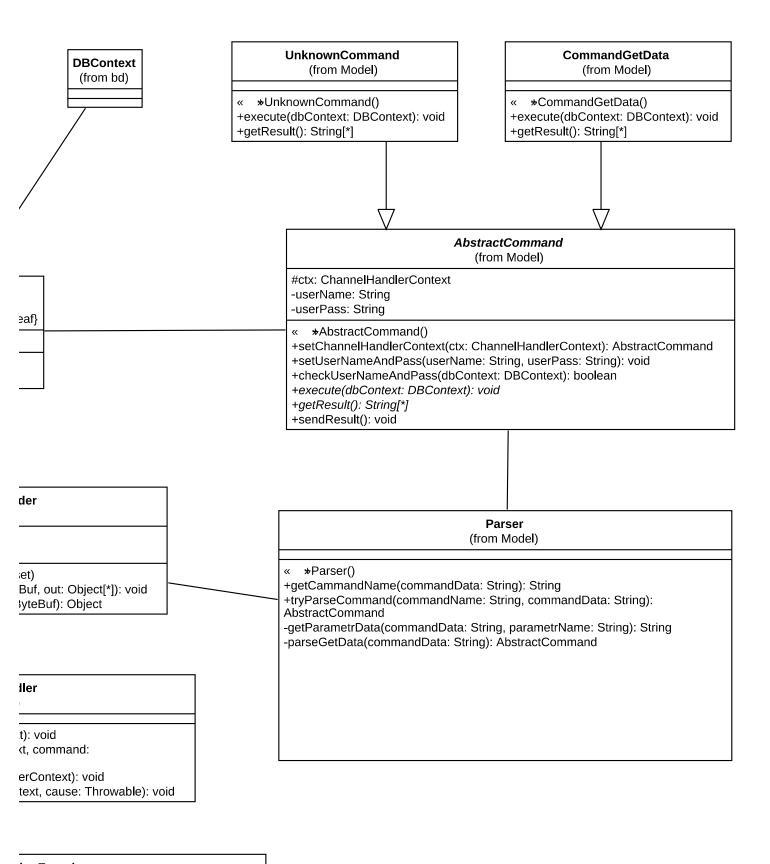
Parameters CommandServer (from protocol) (from protocol) +isSSL: boolean -commandQueue: BlockingQueue {readOnly} +port: int = 5050 -threadPool: ExecutorService {readOnly} +netBossThreads: int = 1 -threadPoolSize: int +netWorkerThreads: int = 2 +commandExecutorThreads: int = 4 *CommandServer(threadPoolSize: int) -initThreadPool(): void *Parameters() +addCommandToProcess(command: AbstractCommand): void +getCommandToDo(): AbstractCommand +stop(): void Server (from protocol) {leaf} CommandExecutor (from protocol) -netServer: NetServer +main(args: String[*]): void ~dbContext: DBContext +start(): void +stop(): void « *CommandExecutor() +getCommandExecutorContext(): CommandServer +run(): void +getNetServer(): NetServer +getParameters(): Parameters **NetServer** (from Model) **BateToCommandDeco** {leaf} (from Model) ~bossGroup: EventLoopGroup -parser: Parser {readOnly} ~workerGroup: EventLoopGroup . -charset: Charset {readOnly} ~sslCtx: SslContext *BateToCommandDecoder(charset: Chars *NetServer() #decode(ctx: ChannelHandlerContext, in: Byte »NetServer(bossThreads: int, workerThreads: int) -decode(ctx: ChannelHandlerContext, buffer: B +ConfigureSSL(SSL: boolean): void +start(PORT: int): void +stop(): void CommandHand (from Model) +channelActive(ctx: ChannelHandlerContex +channelRead0(ctx: ChannelHandlerConte) AbstractCommand): void +channelReadComplete(ctx: ChannelHandle +exceptionCaught(ctx: ChannelHandlerCon NetServerChannelInitializer (from Model) -ENCODER STRING TO BATE: StringEncoder {readOnly} -sslCtx: SslContext {readOnly} CommandToStr »NetServerChannelInitializer(sslCtx: SslContext) (from Mo +initChannel(ch: SocketChannel): void #encode(ctx: ChannelHandlerContext, command



ingEncoder

d: AbstractCommand, out: Object[*]): void