## AgentManager **CmdLine** - m\_control\_mutex: std::mutex - m\_manager: AgentManager& - m\_main\_thread: boost::thread - m\_main\_thread: boost::thread - HELP\_USAGE: static const std::stirng - m\_discover\_port: uint16\_t - m\_server\_port: uint16\_t - cmd\_start(const std::string&, const std::vector<std::string>&): bool - m\_config: Configuration& - cmd\_stop(const std::string&, const std::vector<std::string>&): bool - m\_db: MySqlJdbcConnector - cmd\_filter(const std::string&, const std::vector<std::string>&): bool - m\_acceptor: std::unique\_ptr<boost::asio::ip::tcp::acceptor> - cmd\_proc(const std::string&, const std::vector<std::string>&): bool - m\_io\_service: boost::asio::io\_service + CmdLine(AgentManager&) - m\_connections: std::map<std::string, std::unique\_ptr<socket>> + run(): void - MAX\_BUFFER\_SIZE: static const int + join(): void - addAgentToDb(const std::string&): void - updateAgentStatus(const std::string&, int): bool + AgentManager(uint16\_t, uint16\_t) + connectToDb(): bool + discoverAgents(): void + loadConfiguration(const std::string&): bool socket je skratka pre + run(): void boost::asio::ip::tcp::socket + join(): void + lock(): void + unlock(): void + refreshAgentStatuses(): void Configuration + updateAgentProcesses(const std::string&, bool): bool - m\_db\_url: std::string + ping(const std::string&): bool - m\_db\_user: std::string + sendMessage(const std::string&, const std::string&): bool - m\_db\_password: std::string + recvMessage(const std::string&, nlohmann::json&): bool - m\_db\_name: std::string + isConnected(const std::string&): bool - m\_agent\_update\_interval: unsigned int + addConnection(const std::string&, std::unique\_ptr<socket>): void + parse(const std::string): bool + getAgentlp(const std::string&): std::string + getDbUrl(): const std::string& + getAgents(): std::vector<std::string>> + getDbUser(): const std::string& + getDbPassword(): const std::string& + getDbName(): const std::string& MySqlJdbcConnector + getAgentUpdateInterval(): unsigned int - m\_driver: sql::Driver\* - m\_connection: std::unique\_ptr<sql::Connection> + connect(const Configuration&): bool + tryReconnect(): bool

+ createStatement(): std::unique\_ptr<sql::Statement>

+ prepareStatement(const std::string&): std::unique\_ptr<sql::PreparedStatement>