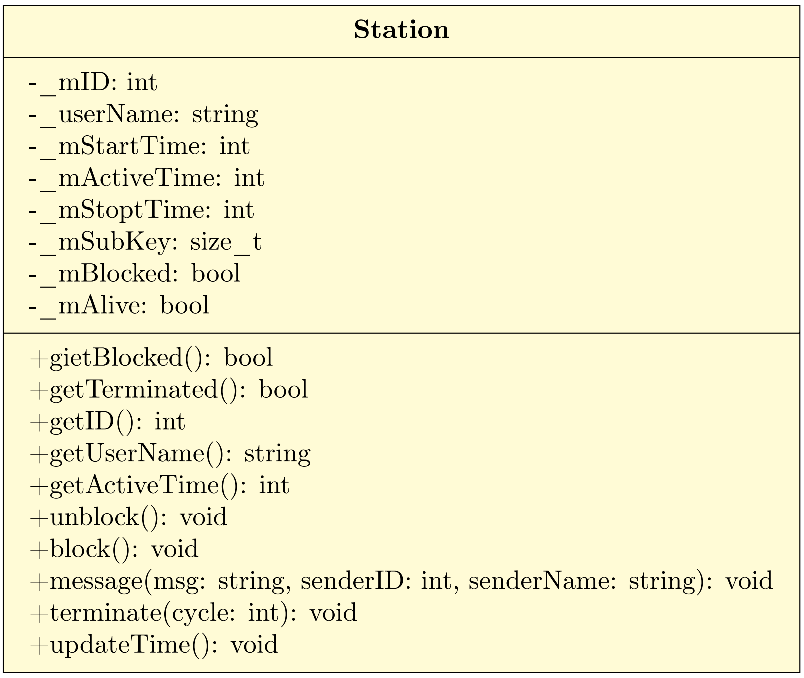
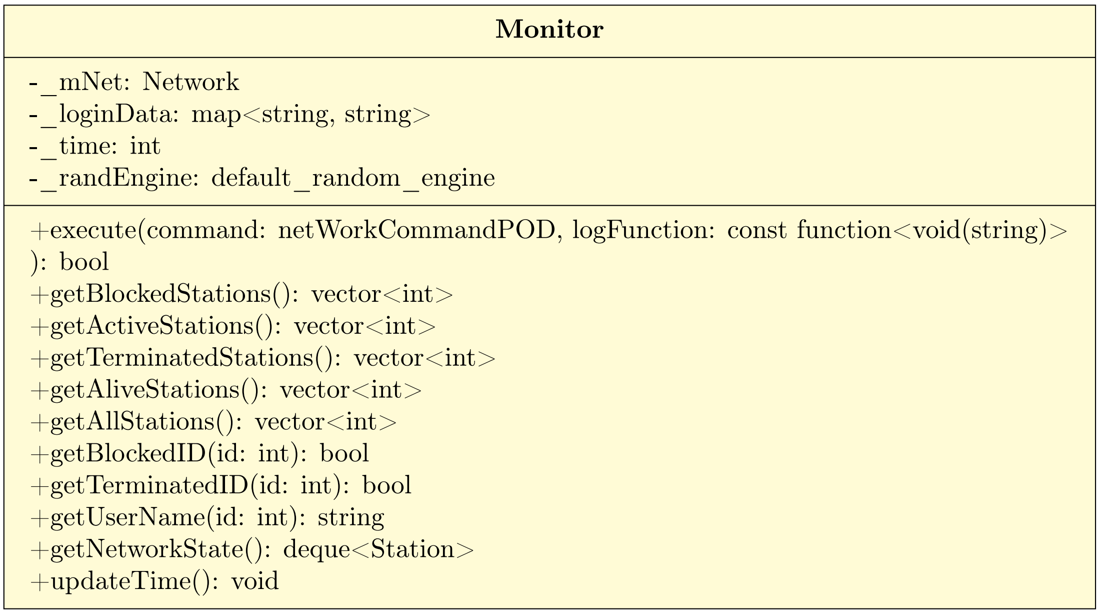


Station

-_mID: int
-_userName: string
-_mStartTime: int
-_mActiveTime: int
-_mStoptTime: int
-_mSubKey: size_t
-_mBlocked: bool
-_mAlive: bool

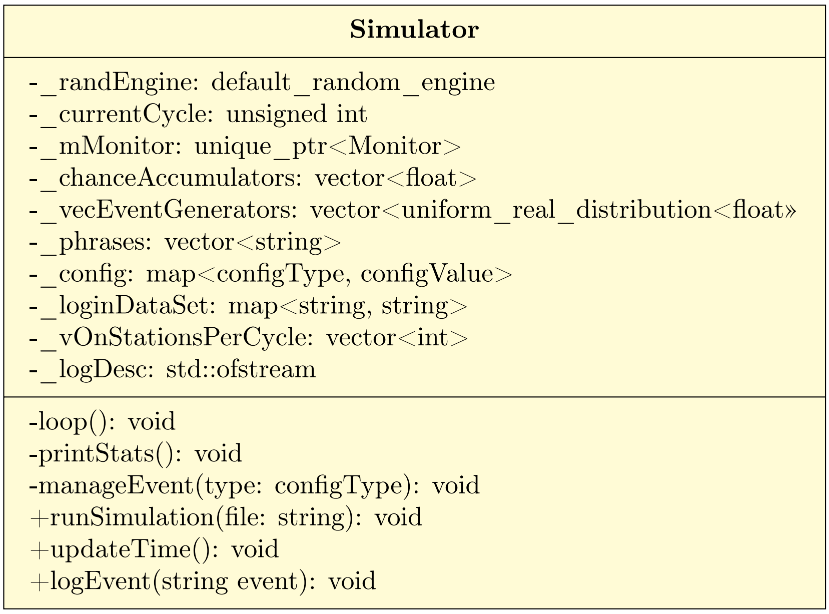
+gietBlocked(): bool
+getTerminated(): bool
+getID(): int
+getUserName(): string
+getActiveTime(): int
+unblock(): void
+block(): void
+message(msg: string, senderID: int, senderName: string): void
+terminate(cycle: int): void
+updateTime(): void



Monitor

-_mNet: Network
-_loginData: map<string, string>
-_time: int
-_randEngine: default_random_engine

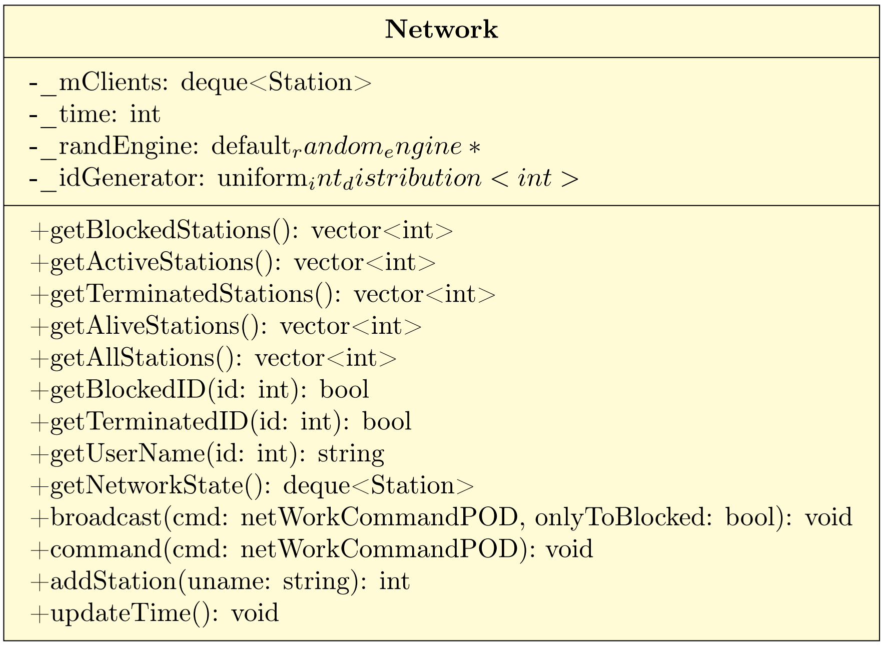
+execute(command: netWorkCommandPOD, logFunction: const function<void(string)>
): bool
+getBlockedStations(): vector<int>
+getActiveStations(): vector<int>
+getTerminatedStations(): vector<int>
+getAliveStations(): vector<int>
+getAllStations(): vector<int>
+getBlockedID(id: int): bool
+getTerminatedID(id: int): bool
+getUserName(id: int): string
+getNetworkState(): deque<Station>
+updateTime(): void



Simulator

-_randEngine: default_random_engine
-_currentCycle: unsigned int
-_mMonitor: unique_ptr<Monitor>
-_chanceAccumulators: vector<float>
-_vecEventGenerators: vector<uniform_real_distribution<float»
-_phrases: vector<string>
-_config: map<configType, configValue>
-_loginDataSet: map<string, string>
-_vOnStationsPerCycle: vector<int>
-_logDesc: std::ofstream

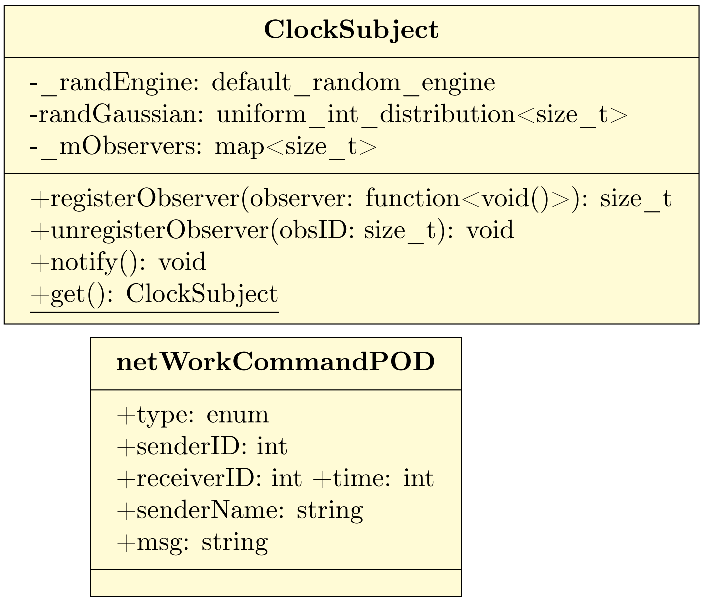
-loop(): void
-printStats(): void
-manageEvent(type: configType): void
+runSimulation(file: string): void
+updateTime(): void
+logEvent(string event): void



Network

-_mClients: deque<Station>
-_time: int
-_randEngine: default$_random_engine*$
-_idGenerator: uniform$_int_distribution<int>$

+getBlockedStations(): vector<int>
+getActiveStations(): vector<int>
+getTerminatedStations(): vector<int>
+getAliveStations(): vector<int>
+getAllStations(): vector<int>
+getBlockedID(id: int): bool
+getTerminatedID(id: int): bool
+getUserName(id: int): string
+getNetworkState(): deque<Station>
+broadcast(cmd: netWorkCommandPOD, onlyToBlocked: bool): void
+command(cmd: netWorkCommandPOD): void
+addStation(uname: string): int
+updateTime(): void


ClockSubject
-_randEngine: default_random_engine
-randGaussian: uniform_int_distribution<size_t>
-_mObservers: map<size_t>
+registerObserver(observer: function<void()>): size_t
+unregisterObserver(obsID: size_t): void
+notify(): void
+get(): ClockSubject
netWorkCommandPOD
+type: enum
+senderID: int
+receiverID: int +time: int
+senderName: string
+msg: string