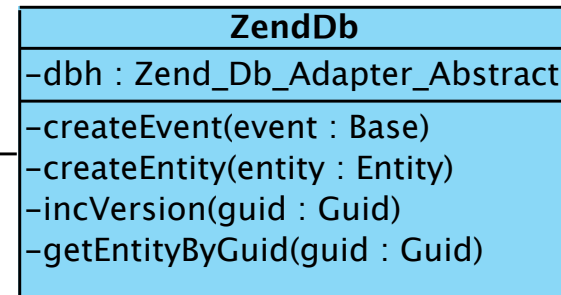
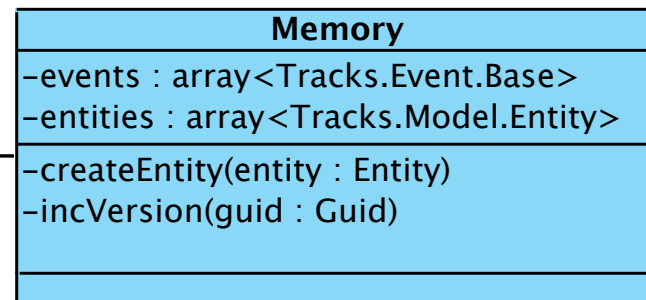
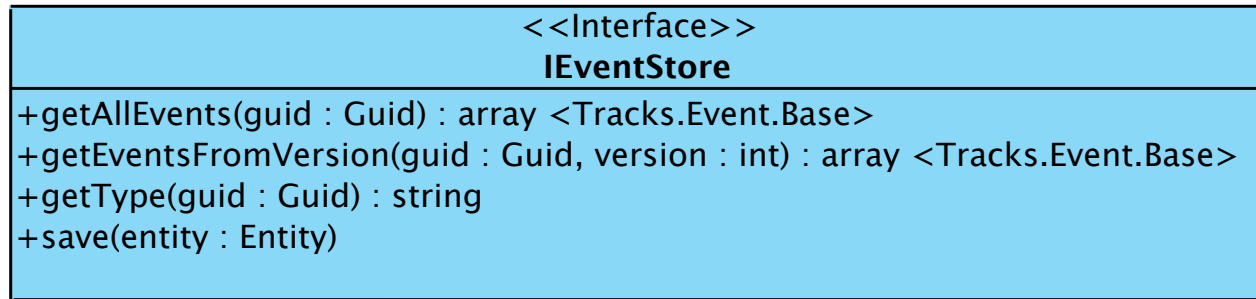


Tracks.EventStore.EventStorage



## Tracks.Model

### Entity

```
#guid : Guid
#deleted : boolean = false
#version : int = 0
#appliedEvents : array<Tracks.Event.Base>
#handlers : array

+loadHistory(history : array <Tracks.Event.Base>)
+getGuid() : Guid
+getVersion() : int
+incVersion(increment : int = 1)
+getAppliedEvents() : array <Tracks.Event.Base>
+getAllAppliedEvents() : array <Tracks.Event.Base>
+clearAppliedEvents()
+clearAllAppliedEvents()
+isCreated() : boolean
+isDeleted() : boolean
+getAllChildEntities() : array <Entity>
+getAllEntities() : array <Entity>
#handleDomainEvent(event : Base)
#applyEvent(event : Base)
#registerEvent(eventName : string, handlerMethod)
-getHandlerName(event : Base) : string
```

### AggregateRoot



### Guid

```
+guid : string

<<Constructor>> +__constructor(guid : string = null)
+__toString() : string
+create() : Guid
+uuid() : string
```

### EntityList

```
-items : array
-guids : array<Tracks.Model.Guid>
-cursor : int = 0

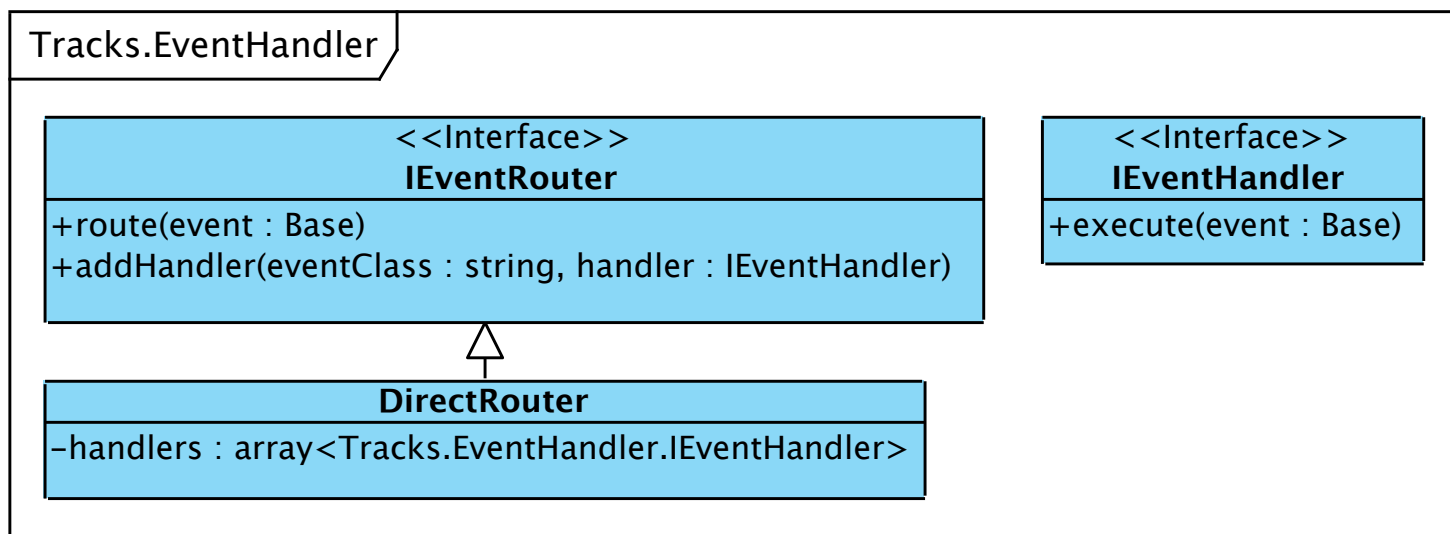
+add(entity : Entity)
+find(guid : Guid) : Entity
+remove(guid : Guid)
+getAllAppliedEvents() : array <Tracks.Event.Base>
+clearAllAppliedEvents()
+getAllChildEntities() : array <Tracks.Model.Entity>
+getAllEntities() : array <Tracks.Model.Entity>
```

Tracks.Event

**Base**

+guid : Guid

<<Constructor>> +\_\_constructor(guid : Guid)  
+getGuid() : Guid



Tracks.Exception

**Base**



**HandlerAlreadyRegistered**

<<Constructor>> +\_\_constructor(domainClass : string, eventClass : string, existingHandler : string)

&lt;&lt;Interface&gt;&gt;

**IEventStore**

+getAllEvents(guid : Guid) : array <Tracks.Event.Base>  
+getEventsFromVersion(guid : Guid, version : int) : array <Tracks.Event.Base>  
+getType(guid : Guid) : string  
+save(entity : Entity)

&lt;&lt;Interface&gt;&gt;

**ISnapshotStore**

+load(guid : Guid) : Entity  
+save(entity : Entity)

**Repository**

-snapshotFrequency : int  
-identityMap : array<Tracks.Model.Entity>  
SNAPSHOT\_FREQUENCY : int = 100

<<Constructor>> +\_\_constructor(eventStore : IEventStore, router : IEventRouter, snapshotStore : ISnapshotStore)  
+load(guid : Guid) : AggregateRoot  
+setSnapshotFrequency(numEvents : int = Tracks.EventStore.Repository.SNAPSHOT\_FREQUENCY)  
+save(aggregateRoot : AggregateRoot)  
-storeInIdentityMap(entity : Entity)  
-loadFromIdentityMap(guid : Guid) : Entity  
-loadFromHistory(guid : Guid, entity : Entity = null) : Entity  
-loadEntity(guid : Guid, entity : Entity = null) : Entity  
-routeEvents(aggregateRoot : AggregateRoot)  
-loadFromSnapshot(guid : Guid) : Entity  
-updateVersionsAndClearEvents(aggregateRoot : AggregateRoot)  
-saveSnapshots(aggregateRoot : AggregateRoot)  
-saveSnapshot(entity : Entity)

EventStorage

SnapshotStorage

## Tracks.EventStore.SnapshotStorage

