Custom Entity Import Export

To add export and import logic to a custom entity, *Export* and *Import Handlers* need to be declared and implemented for the Entity.

As an example, let's add importer and exporter for a hypothetical Leaderboard Entity.
First we declare the entity as import/exportable by adding `[EntityArchiveImporterExporter]` on the `EntityCondfig` of the Entity. For example:

```
// Register tag "leaderboard" for the Archive
// Set ImportLeaderboardHandler as importer
// Set ExportLeaderboardHandler as exporter
[EntityArchiveImporterExporter("leaderboard", typeof(ImportLe
aderboardHandler), typeof(ExportLeaderboardHandler))]
public abstract class LeaderboardEntityConfigBase : Persisted
EntityConfig
{
    ...
```

Then we add implement the handlers. To export as persisted entity, we can implement a minimal handler with `SimpleExportEntityHandler`:

```
public class ExportLeaderboardHandler : SimpleExportEntityHan
dler
{
    public ExportLeaderboardHandler() : base(EntityKindGame.L
eaderboard)
    {
    }
}
```

And to import, we can use `SimpleImportEntityHandler` as follows:

```
public class ImportLeaderboardHandler : SimpleImportEntityHan
dler
{
    public ImportLeaderboardHandler() : base(EntityKindGame.L
eaderboard)
    {
    }

    protected override void AssignBasicRuntimePropertiesToMod
el(IModel model)
    {
        // Example, set GameLogic and LogicVersion. This shou
ld be similar to AssignBasicRuntimePropertiesToModel
        // of the Actor.
        LeaderboardModel leaderboardModel = (LeaderboardMode
l)model;
        leaderboardModel.GameConfig = ActiveGameConfig.Baseli
neGameConfig.SharedConfig;
        leaderboardModel.LogicVersion = LogicVersion;
    }

    protected override async Task DoRemapModel(IModel model,
EntityImportEntityIdRemapper remapper)
    {
        // This remaps, i.e. updates the EntityId references
contained in the model.
        // For leaderboard, we could for example update conta
ined players' ids.
        LeaderboardModel leaderboardModel = (LeaderboardMode
l)model;
        foreach (Player in leaderboard.Players)
           player.Id = await remapper.Get(player.Id);
    }

    protected override await Task DoWrite(EntityId id, bool c
```

```
reateFirst, IEntityAsker asker)
    {
        // This method should write the import the entity. Co
mmon pattern is to
        // first create an empty entity record in the Databas
e, and then EntityAsk
        // the Entity itself to import data. For example.

        if (createFirst)
        {
            await DatabaseEntityUtil.CreateNewEntityAsync<Per
sistedLeaderboard>(entityId)
        }

        LeaderboardImportModelDataRequest request = new Leade
rboardImportModelDataRequest(Payload, SchemaVersion, isNewEnt
ity: createFirst);
        LeaderboardImportModelDataResponse response = await a
sker.EntityAskAsync<LeaderboardImportModelDataResponse>(entit
yId, request);
        if (!response.Success)
        {
            throw new ImportException($"Overwrite of leaderbo
ard {entityId} failed", response.FailureInfo);
        }
    }
}
```

Where the message are roughly:

```
[MetaMessage(123, MessageDirection.ServerInternal)]
public class LeaderboardImportModelDataRequest : MetaMessage
{
    public byte[] Payload { get; private set; }
    public int? SchemaVersion { get; private set; }
```

```
    LeaderboardImportModelDataRequest() { }
    public LeaderboardImportModelDataRequest(byte[] payload,
int? schemaVersion)
    {
        Payload = payload;
        SchemaVersion = schemaVersion;
    }
}

[MetaMessage(124, MessageDirection.ServerInternal)]
public class LeaderboardImportModelDataResponse : MetaMessage
{
    public bool Success { get; private set; }
    public string FailureInfo { get; private set; }

    LeaderboardImportModelDataResponse() { }
    public LeaderboardImportModelDataResponse(bool success, s
tring failureInfo) { Success = success; FailureInfo = failure
Info; }
}
```

Finally, we need to implement the actual importing in the actor.

```
[EntityAskHandler]
async Task<LeaderboardImportModelDataResponse> HandleLeaderbo
ardImportModelDataRequest(LeaderboardImportModelDataRequest r
equest)
{
    // Try to deserialise the incoming model
    try
    {
        // Deserialize model
        LeaderboardModel importModel = DeserializePersistedPa
yload(request.Payload, _baselineGameConfigResolver, _logicVer
sion);
```

```
        // Run migrations
        if (request.SchemaVersion.HasValue)
            MigrateState(importModel, request.SchemaVersion.V
alue);

        // Perform some other post processing here

        // Set new model
        SwitchToNewModelImmediately(importModel);

        // Example: Kick all clients
        foreach (EntitySubscriber subscriber in new List<Enti
tySubscriber>(_subscribers.Values))
        {
            if (subscriber.Topic == EntityTopic.Member)
                KickSubscriber(subscriber, null);
        }

        // Persist before responding ok
        await PersistStateIntermediate();

        return new LeaderboardImportModelDataResponse(true, n
ull);
    }
    catch (Exception ex)
    {
        return new LeaderboardImportModelDataResponse(false,
ex.ToString());
    }
}
```