

Contents

- [EntityManager overview](#)
- [What next?](#)

The [EntityManager](#) is the gateway to the persistence service and holds a cache of entities that the application is working with, including entities that have been queried, added, updated, and marked for deletion. The EntityManager is the core class in DevForce, and this page discusses its primary capabilities.

EntityManager overview

The *EntityManager* serves three main functions:

1. It communicate with the application server.
2. It queries and saves entities.
3. It holds entities in a local container called the [entity cache](#).

When the client application requires data, it typically calls a method on an instance of an *EntityManager*. The *EntityManager* establishes communication channels, sets up the client's security context, serializes and deserializes data, and regulates the application-level flow of traffic with the persistence service on the [EntityServer](#).

Query and save requests are the predominant *EntityManager* activities. You can ask the *EntityManager* to [query for entities](#) in a wide variety of ways:

- [with LINQ queries](#)
- [by EntityKey](#)
- [with Entity SQL \(ESQL\)](#)
- [with stored procedure queries](#)
- [by navigating](#) from one entity to another (e.g., aCustomer.Orders)

However you do it, *EntityManager* puts the queried entity results in its [entity cache](#). When you create new entities, you add them to that cache. When you delete entities, you are actually marking entities in the cache, scheduling them to be deleted. The inner workings of changing entities in cache is covered under the [Create, modify and delete](#) topic.

When you eventually ask the *EntityManager* to [save](#), it finds the changed entities in cache - the ones you've modified, added, and scheduled for deletion - and sends their data to the *EntityServer*. If all goes well, the *EntityServer* reports success and the EntityManager adjusts the cached entities to reflect the save by discarding deleted entities and re-setting the [EntityState](#) of the added and modified entities to *Unchanged*.

What next?

Drill deeper in this topic for the next level of detail about the *EntityManager*.