


Improve Entity Framework Performance



 Bulk Insert

 Bulk Delete

 Bulk Update

 Bulk Merge

[LEARN MORE](#)

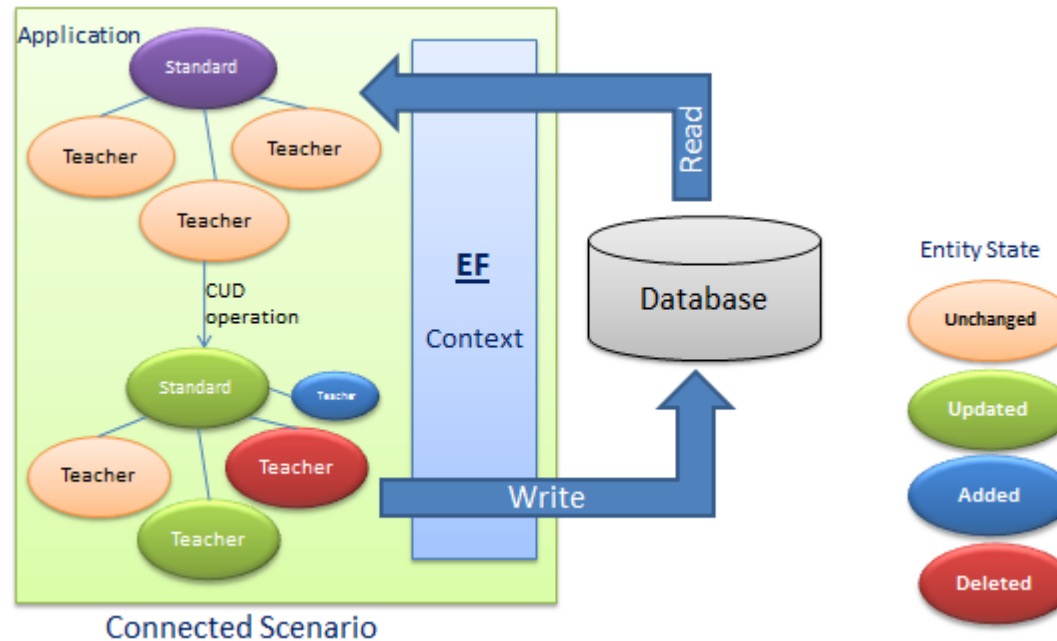
[< Previous](#)[Home](#)

Persistence in Entity Framework

There are two scenarios when persisting (saving) an entity to the database using Entity Framework: the Connected Scenario and the Disconnected Scenario.

Connected Scenario

In the connected scenario, the same instance of the context class (derived from DbContext) is used in retrieving and saving entities. It keeps track of all entities during its lifetime. This is useful in windows applications with the local database or the database on the same network.



Pros:

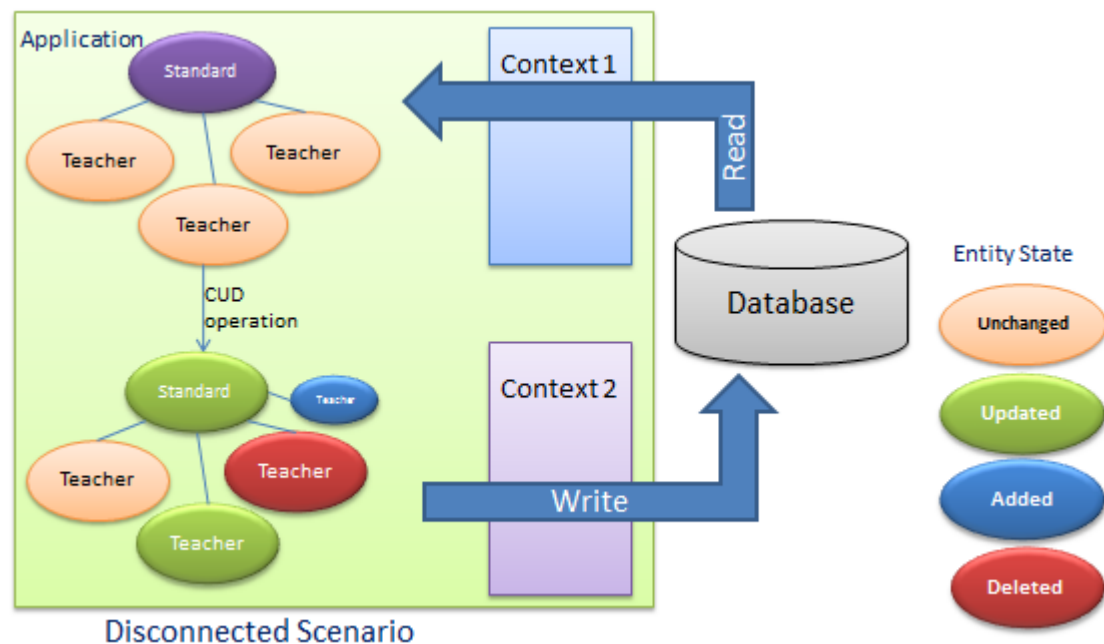
- Performs fast.
- The context keeps track of all entities and automatically sets an appropriate state as and when changes occur to entities.

Cons:

- The context stays alive, so the connection with the database stays open.
- Utilizes more resources.

Disconnected Scenario

In the disconnected scenario, different instances of the context are used to retrieve and save entities to the database. An instance of the context is disposed after retrieving data and a new instance is created to save entities to the database.



The disconnected scenario is complex because an instance of the context doesn't track entities, so you must set an appropriate state to each entity before saving entities using `SaveChanges()`. In the figure above, the application retrieves an entity graph using Context 1 and then the application performs some CUD (Create, Update, Delete) operations using Context 2. Context 2 doesn't know what operation has been performed on the entity graph in this scenario.

This is useful in web applications or applications with a remote database.

Pros:

- > Utilizes less resources compared to the connected scenario.
- > No open connection with the database.

Cons:

- > Need to set an appropriate state to each entity before saving.
 - > Performs slower than the connected scenario.
-

[< Previous](#)[Home](#)

ENTITYFRAMEWORKTUTORIAL

Learn Entity Framework using simple yet practical examples on EntityFrameworkTutorial.net for free. Learn Entity Framework DB-First, Code-First and EF Core step by step. While using this site, you agree to have read and accepted our terms of use and [privacy policy](#).

✉ feedback@entityframeworktutorial.net

TUTORIALS

[> EF Basics](#)[> EF 6 DB-First](#)[> EF Core](#)[> EF 6 Code-First](#)

E-MAIL LIST

Subscribe to EntityFrameworkTutorial email list and get EF 6 and EF Core Cheat Sheets, latest updates, tips & tricks about Entity Framework to your inbox.

We respect your privacy.

[HOME](#) [PRIVACY POLICY](#) [ADVERTISE WITH US](#)

© 2020 EntityFrameworkTutorial.net. All Rights Reserved.