

УДК 004.02

## ИНТЕГРАЦИЯ ENTITY FRAMEWORK И СПЕЦИФИКАЦИИ GRAPHQL

**Масалкин С.Е.** (федеральное государственное автономное образовательное учреждение высшего образования “Национальный исследовательский университет ИТМО”)

**Научный руководитель – кандидат технических наук, доцент Маркина Т.А.**  
(федеральное государственное автономное образовательное учреждение высшего образования “Национальный исследовательский университет ИТМО”)

**Аннотация.** Бесшовная и однозначная трансляция GraphQL запросов в структуры IQueryable и Expression, которые распознаёт провайдер Entity Framework. Корректная проекция, фильтрация и возможность соединения связанных сущностей на уровне выполнения. Решение проблемы N+1 при параллельном вычислении GraphQL схемы.

**Введение.** Спецификация GraphQL является одной из самых популярных альтернатив для REST. За счёт своей гибкости она отлично подходит для мастер-модулей в микросервисной архитектуре. Клиент на основе схемы сам определяет поля, которые ему необходимы. Однако на сегодняшний день не все существующие движки, которые реализуют спецификацию под платформу .NET, могут бесшовно и однозначно транслировать запросы GraphQL в запросы к базе данных. К примеру, в фреймворке GraphQL.NET для вычисления типа необходимо явно включать кодовую базу, которая будет составлять Entity Framework запрос. Существует фреймворк HotChocolate, который позволяет более гибко настраивать движок. Он способен определить типы, которые запрашивает клиент и построить простой запрос к базе данных. Однако при выполнении запроса нет возможности сделать проекцию или фильтрацию модели базы данных.

**Основная часть.** Необходимо выбрать фреймворк, который реализует спецификацию GraphQL под платформу .NET. Кандидатом является фреймворк HotChocolate, потому что он открыт для расширения движка, у него большое сообщество. Расширить функционал движка можно в виде NuGet пакета. Ввиду того, что поля запроса GraphQL выполняются параллельно, это создает проблему использования одного Unit-Of-Work контекста на запрос. Entity Framework исключает возможность параллельного доступа нескольких потоков к одному экземпляру контекста. В виде решения может помочь технология context-pooling. Чтобы обеспечить проекцию полей модели базы данных, можно в процесс выполнения запроса GraphQL внедрить функционал, который сможет динамически, ориентируясь на схему, составлять LINQ Expression. В этом может помочь шаблон разработки Middleware. Фильтрация модели данных будет построена по такому же принципу. Также при выполнении Many-to-One GraphQL запроса может возникнуть проблема N+1. Может возникнуть ситуация, в которой для каждой дочерней сущности будет сделан запрос для поиска родительской. Чтобы избежать такую проблему можно воспользоваться технологией Facebook DataLoader. С помощью неё можно будет собирать ключи родительских сущностей и в конце выполнить один запрос к базе данных.

**Выводы.** Решение будет внедрено в один из проектов компании “АО Аркадия”, который посвящен HR&Recruitment. Также решение можно будет собрать в отдельный NuGet пакет как расширение фреймворка GraphQL.

Масалкин С.Е. (автор)

Подпись

Маркина Т.А. (научный руководитель)

Подпись