

Выполнил(а) Васильченко Роман Антонович, № группы P3117, оценка _____
Фамилия И.О. студента не заполнять

| | | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|-----------------------------------------------|
| Название статьи/главы книги/видеолекции gRPC vs REST, что выбрать для нового сервера? | | |
| ФИО автора статьи (или e-mail) <u>deft31</u> | Дата публикации (не старше 2018 года) "29" июня 2021 г. | Размер статьи (от 400 слов) 2418 |
| Прямая полная ссылка на источник и сокращённая ссылка (bit.ly, goo.gl, tr.im и т.п.) https://habr.com/ru/post/565020/ https://bit.ly/3DjThaM | | |
| Теги, ключевые слова или словосочетания REST Сервер HTTP gRPC Скорость работы API Монолит и микросервис | | |
| Перечень фактов, упомянутых в статье 1. В настоящий момент REST намного более популярен, чем gRPC 2. Json является самым главным форматом обмена информации в REST 3. gRPC написан Гугл и использует свой формат обмена сообщений ProtoBuf, который имеет высокую степень упаковки для быстрой сериализации / десериализации данных, чем JSON 4. Монолиты проще писать, но со временем их тяжелее изменять. Микросервисы имеют противоположную проблему. 5. У gRPC существует компилятор protoc, который работает с множеством языков. | | |
| Позитивные следствия и/или достоинства описанной в статье технологии (минимум три пункта) 1. gRPC построен на HTTP 2.0, когда REST на 1.1 2. Скорость передачи сообщений быстрее в 7-10 раз 3. Скорость получается путем глубокой упаковки Protobuf, что приводит к тому, что можно сосредоточиться только на сериализации и десериализации данных 4. Protoc позволяет удобно работать в системе из множества языков | | |
| Негативные следствия и/или недостатки описанной в статье технологии (минимум три пункта) 1. gRPC и ProtoBuf намного сложнее использовать, а также сложнее найти примеры кода с ними из-за их малой популярности 2. Из-за высокой упаковки Ptotobuf менее читаемый, чем JSON 3. Из-за универсальности REST gRPC может быть эффективно использован только во внутренних системах (инфраструктура, которая закрыта от внешних пользователей) или в сервисах, где каждая наносекунда важна (относительно мало компаний) | | |