

В. Н. Леванцов, А. С. Чеботаревский

(ГГУ им. Ф. Скорины, Гомель)

GLOBALS DB – УНИВЕРСАЛЬНАЯ NOSQL БАЗА

Globals DB представляет собой бесплатную базу данных, отличительной особенностью которой является использование в качестве механизма сохранения данных Глобальных Постояннохраняемых Переменных (Global Persistent Variables).

В качестве движка в GlobalDB используется такой же движок как и БД Cache, поэтому в каком-то из смыслов GlobalDB представляет собой бесплатную реализацию Cache. Cache прошел через этапы длительной разработки (в таких сферах как, финансы, здравоохранение, банковское дело), отладки и тестирования, в итоге получил скорость работы как у баз размещаемым в ОЗУ и целостность данных на уровне баз работающих с жесткий диск. Cache обладает гораздо большим функционалом, но из-за одинакового ядра приложения для GlobalsDB можно без всяких изменений перенести на Cache.

Варианты обращения к GlobalsDB, через высокопроизводительный внутрипроцессный API, разделился между Node.JS (Javascript), .NET и Java. В качестве ускорения работы GlobalDB было принято решение напрямую обращаться к программному коду из процесса приложения, а не использовать сокеты и порты. Для получения еще большей скорости GlobalDB привязывается к процессу приложения.

В терминах GlobalsDB глобалы — это структуры для хранения данных.

Глобалы представляют собой переменные, значения которых сохраняются на жесткий диск. Эти значения есть ни что иное как абстракция над деревьями (B-trees). Так же они могут быть использованы для хранения больших объемов данных представленных как многомерные разреженные массивы. С помощью глобалов можно очень качественно и высокоэффективно решать задачи, для решения которых принято использовать различные NoSQL-базы (например, тот же MongoDB).

Глобалы в отличие от таблиц, не нуждаются в определении (CREATE TABLE) прежде чем с ними можно начать работу. Они не имеют заранее заданной структуры (schema-free). Также глобалы иерархически структурированы (речь о внутренней структуре каждого отдельного глобала), разреженные, динамические. Глобалы очень похожи на ассоциативные массивы. Пример объявления глобалов:

```
myTable["101-22-2238", "Chicago", 2] = "Some information"
```

```
account["New York", "026002561", 35120218433001] = 123456.45
```

Каждый элемент глобала, определяется именем глобала и сочетанием индексов. Все элементы глобалов хранят текстовый строку, естественно можно хранить и пустые строки. Глобалы позволяют динамически создавать, изменять и удалять себя, без всякого объявления или определения схем данных. Таким образом получается, что все схемы данных и все высокоуровневые абстракции находятся в голове у разработчика.

Главной причиной экстремально высокой производительностью GlobalDB, является интеллектуальный кэширующий механизм. Суть его заключается в том, что в большинстве случаев, когда мы хотим получить доступ к элементам глобала, то они уже находятся в оперативной памяти. Этот механизм улучшался и дорабатывался в Cache на протяжении многих лет. Поэтому он и показывает настолько выдающиеся результаты.

Если кратко охарактеризовать GlobalDB, то к плюсам можно отнести: эластичное масштабирование; работа с гигантскими объемами данных с экстремальной производительностью; системы на глобалах могут работать безо всякого присмотра десятилетиями; экономичность; гибкие модели данных; состоятельность и зрелость технологии; поддержка; аналитика и бизнес-аналитика.

К ярко выраженным минусам можно отнести низкое число экспертов. Это единственная область, где позиции GlobalDB слабы. Оно обуславливается тем, что число опытных профессионалов по сравнению с теми же профессионалами работающими с реляционными СУБД значительно меньше. Бесплатная реализация GlobalDB должна привлечь пользователей как к самой GlobalDB, так и к ее предку, а именно к Cache.

Если подвести итог, то можно с уверенностью сказать, что GlobalDB, по сумме своих характеристик становится идеальным кандидатом в тех случаях, когда требуется надежная, взрослая и быстрая NoSQL база. Большую роль в возникшей популярности GlobalDB играет простой переход на Cache, который дает доступ к дополнительным возможностям и инструментариям, а также к высокотехнологично функциональности NoSQL БД.

Одним из самых оптимальных вариантов использования GlobalDB – это использовать данную его вместе с очень быстрой и экстремально высокой производительностью Node.js. В этом случае мы получим идеальную, мощную высокопроизводительную промышленную платформу для реализации NoSQL баз данных.