

## GQL MANIFESTO

*It seems like the time is right to create one standard property graph query language. Fusing the best of Cypher, PGQL and G-CORE into a more comprehensive query language built specifically for graph solutions.*

*Should the property graph community unite to create a standard graph query language, GQL, alongside SQL? [Vote now](#) or read on and vote afterwards.*

### **Property graph data has a big presence**

*Amazon Neptune, Oracle PGX, Neo4j Server, SAP HANA Graph, AgensGraph (over PostgreSQL), Azure CosmosDB, Redis Graph, SQL Server 2017 Graph, Cypher for Apache Spark, Cypher for Gremlin, SQL Property Graph Querying, TigerGraph, Memgraph, JanusGraph, DSE Graph ... it's hard to keep up.*

### **And an even bigger future**

*Thousands of applications already use the rich, intuitive graph data model. The ability to express, find and extract complex relationships and patterns; to execute graph algorithms; to register transactions; and to perform graph analysis is proving its value every day, in every domain of business and science. And adoption is accelerating.*

### **Relational data has SQL**

*SQL is one of the key underpinnings of modern information technology. We need a declarative query language for the powerful – and distinct – property graph data model, to*

## МАНИФЕСТ GQL

Похоже, что настало время для создания единого стандартного языка запросов для графов. Объединив лучшее из Cypher, PGQL и G-CORE в более понятный язык запросов, построенный специально для графовых решений.

Должно ли графовое сообщество объединиться для создания стандартного языка запросов для графов (GQL), по подобию SQL? [Проголосуйте](#) сейчас или после прочтения этого манифеста.

### **Графовых баз данных очень много**

Amazon Neptune, Oracle PGX, Neo4j Server, SAP HANA Graph, AgensGraph (на PostgreSQL), Azure CosmosDB, Redis Graph, SQL Server 2017 Graph, Cypher for Apache Spark, Cypher for Gremlin, SQL Property Graph Querying, TigerGraph, Memgraph, JanusGraph, DSE Graph ... тяжело уследить за всеми.

### **И у них большие перспективы**

Тысячи приложений уже используют богатую, интуитивно понятную графовую модель данных. Возможность выражать, находить и извлекать сложные структуры и связи, запускать графовые алгоритмы, регистрировать транзакции и анализировать графы каждый день доказывает свою пользу во всех сферах бизнеса и науки. И распространение всё увеличивается.

### **Для реляционной модели данных существует SQL**

SQL является одной из основ современных информационных технологий. Для этой же цели нам требуется декларативный язык запросов

*play a similar role.*

*Many graph databases and services are not features of relational systems, but native, optimized implementations. And many graph queries, when implemented over relational systems, do not need to be tied to tables on the surface. There is a big role for graph extensions to SQL, but they are unlikely to be the leading edge of graph querying.*

### **Property graph data needs GQL**

*Like SQL, the new GQL (Graph Query Language) needs to be an industry standard.*

*Different languages for different products help no one. An international standard will let the whole graph data market grow, to the mutual benefit of all vendors and all users. One language, one skill set. A common query language focuses support around data modelling, ETL and visualization tools for graph data, and portable queries mitigate vendor lock-in.*

*But GQL also needs to be tuned and agile to meet the needs of the expanding property graph data industry. It should work with SQL, but it should not be confined by SQL. GQL would be a language that complements the traversal API of Apache Tinkerpop's Gremlin as well as SPARQL for RDF triple-stores. This results in better choices for developers, data engineers, data scientists, CIOs and CDOs alike.*

для графовой модели данных, такой же значимый и понятный.

Многие графовые базы данных и сервисы созданы и работают нативно и оптимизировано для графов, а не как часть реляционной системы. Также многие запросы к графам, работающим в реляционных системах, не обязаны быть привязанными к таблицам. Графовые расширения для SQL играют большую роль, но вряд ли будут ключевым звеном в работе с графами.

### **Для графовой модели данных необходим GQL**

Как и SQL, новый GQL (Graph Query Language, язык графовых запросов) должен стать стандартом для индустрии.

Разные языки для разных продуктов не помогут никому. Международный стандарт позволит расти всему рынку графовых баз данных, с выгодой для всех вендоров и пользователей. Один язык, один набор навыков. Общий язык, сфокусированный на моделировании, ETL (Extract, Transform, Load - "извлечение, преобразование, загрузка"), а также инструментах визуализации данных и стандартизированных запросов, не привязанных к вендору.

GQL должен быть отлаженным и достаточно гибким для удовлетворения потребностей растущей индустрии графовых данных. Он должен уметь работать с SQL, но не должен быть ограничен SQL. GQL станет языком, который дополняет как Gremlin, язык обхода графов от Apache TinkerPop, так и SPARQL для хранилищ RDF-триплетов. Результатом станет лучший выбор для разработчиков, инженеров и аналитиков данных, CIO и CDO.

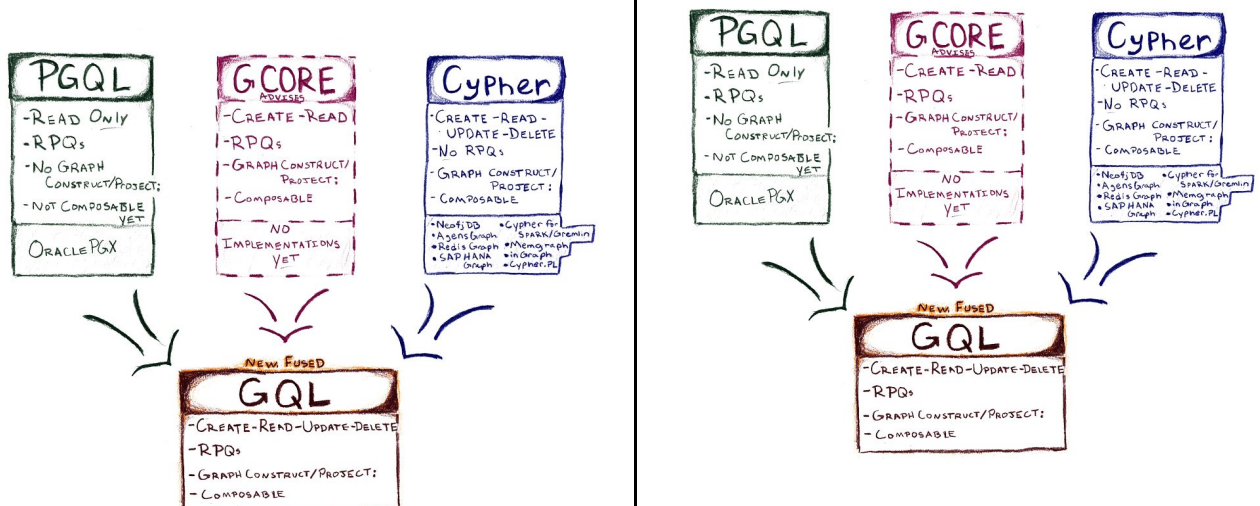
## Three languages

Right now there are three property graph query languages that are closely related.

We have Cypher (from Neo4j and the openCypher community). We have PGQL (from Oracle). And we have G-CORE, a research language proposal from Linked Data Benchmark Council (co-authored by world-class researchers from the Netherlands, Germany, Chile, the U.S, and technical staff from SAP, Oracle, Capsenta and Neo4j).

You'd be hard pressed to tell them apart for many common graph queries. They have very similar data models, syntax and semantics. Each one is ahead of the game or behind the game in one way or another.

Their authors share many ambitions for the next generation of graph querying such as a composable graph query language with graph construction, views and named graphs; and a pattern-matching facility that extends to regular path queries, where regular expressions find and retrieve patterns even more concisely and flexibly.



## Три языка

Сейчас существуют три тесно связанных между собой языка запросов для графов.

У нас есть Cypher (от Neo4j и сообщества openCypher). У нас есть PGQL (от Oracle). И у нас есть G-CORE, новый исследовательский язык, предлагаемый Linked Data Benchmark Council (при соавторстве исследователей мирового уровня из Нидерландов, Германии, Чили, США и технического персонала из SAP, Oracle, Capsenta и Neo4j).

Многие запросы на этих языках повторяют друг друга. У всех них очень похожие модели данных, синтаксис и семантика. Каждый язык в чём-то лучше или хуже.

Их авторы согласны во многом касательно того, как должны выглядеть графовые запросы в скором будущем: композиция запросов, конструирование графов, именованные графы; а также механизмы паттерн-матчинга, где гибкий поиск путей в графе осуществляется регулярными выражениями.

### **Three into one should go**

*My name is Alastair Green. I help to organize work on graph query language standards and research at Neo4j, Inc., working with talented engineers, language architects and standards specialists within our company and among collaborators in industry and academia.*

*Sponsored by our CEO, Emil Eifrem, and our VP of Products, Philip Rathle, the Neo4j team is advocating that the database industry and our users get away from two or three blurred photocopies of the same ideas, and get together to define and standardize onelanguage.*

*Bringing PGQL, G-CORE and Cypher together, we have a running start. Two of them are industrial languages with thousands of users, and combined with the enhancements of a research language, they all share a common heritage of ASCII art patterns to match, merge and create graphs.*

### **One property graph query language, one GQL**

*We'll be working with anyone who wants to help create GQL as a new open industry standard. We'd love it to be an ISO standard alongside SQL. Or perhaps it will end up in another venue.*

*But that's not the most important thing. A technically strong standard, with strong backing among vendors and users: that's what matters most. So we'll be appealing for your public, vocal support. If there is a will,*

### **Три должны объединиться в один**

Меня зовут Аластер Грин (Alastair Green). Я занимаюсь организацией работы над стандартами графовых языков и исследований в Neo4j, Inc., где я работаю с талантливыми инженерами, языковыми архитекторами и специалистами по стандартам внутри нашей компании и среди единомышленников в индустрии и академической среде.

При поддержке Эмиля Эйфрема (Emil Eifrem), нашего CEO, и Филипа Ратла (Philip Rathle), вице-президента по продуктам, команда Neo4j призывает индустрию баз данных и наших пользователей освободиться от двух или трёх тусклых ксерокопий одинаковых идей и объединиться с целью создания и стандартизации одного языка.

Объединение PGQL, G-CORE и Cypher — это хорошее начало. Два из них признаны тысячами пользователей в индустрии. И, вместе с исследовательским языком, все они разделяют общее наследие ASCII-арт паттернов для поиска, изменения и создания графов.

### **Один язык запросов для графовой модели данных, один GQL**

Мы готовы сотрудничать с каждым, кто хочет помочь сделать GQL новым открытым стандартом в индустрии. Мы хотели бы, чтобы наравне с SQL это был стандарт ISO, а может и какой-то другой.

Но это не самое главное. Гораздо важнее, чтобы этот стандарт был технически мощным, с сильной поддержкой среди вендоров и пользователей. Поэтому мы призываем к громкой публичной поддержке. Было бы желание, а возможность найдется.

*there will be a way.*

*Open letter to the database industry*

*Vote Now*

*Should the property graph community unite to create a standard Graph Query Language, GQL, alongside SQL?*

*YesNo*

*Vote*

*View Results*

*Results*

Открытое письмо к индустрии баз данных (на английском)

Голосуйте сейчас

Должно ли сообщество графовых баз данных объединиться для создания стандартного языка графовых запросов GQL (Graph Query Language), по подобию SQL?

Да Нет

Голосовать

Посмотреть результаты

Результаты