



# Inside the Standardization Machine Room

## How ISO/IEC 39075:2024 - GQL was produced

Hannes Voigt, 17th LDBC TUC Meeting, Santiago de Chile, June 09 2024

# Who am I? — Past

- 2008–2018 at TU Dresden
- 2010–2011 SAP Labs, Palo Alto  
Query language design for a predecessor project of SAP HANA Graph
- Since 2015 Member of LDBC Language Task Force  
Resulted in the G-Core query language design
- Since 2017 Involved in openCypher  
Group of Cypher implementers discussing query language design around Cypher

# Who am I? – Present

- Since 2018 at Neo4j, Query Languages Standards and Research Team
  - Design work for ISO/IEC standardization project 39075 GQL
  - Design work around Cypher for the Neo4j products
- Chair of the Incits DM GQL Expert Group (US level)
  - Group were the majority of involved vendors discuss design work for GQL
- Member of ISO/IEC JTC1 SC32 WG3 (international level)
  - Committee responsible for the ISO/IEC 9075 SQL standard
  - Home of the ISO/IEC 39075 GQL project
- Member of the LDBC Property Graph Schema Working Group (PGS WG)
  - LDBC GQL Community Group studying designs for property graph schema

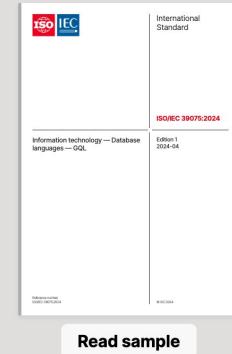
# GQL?



# GQL

[<https://www.iso.org/standard/76120.html> and <https://www.gqlstandards.org/>]

- ISO standard: ISO 39075
- Defines “a database language for modeling structured data as a graph, and for storing, querying, and modifying that data in a graph database or other graph store”
- “GQL addresses the Property Graph model.”
- Developed by ISO/IEC JTC1 SC32 WG3 – the “SQL committee”
- Published in April 2024



ISO/IEC 39075:2024

Information technology — Database languages — GQL

Published (Edition 1, 2024)

A screenshot of a product page for ISO/IEC 39075:2024. At the top right, the title "ISO/IEC 39075:2024" is displayed. To the right of the title is a large number "216" with a "CHF" symbol above it, indicating the price in Swiss francs. Below the title, there are dropdown menus for "Format" (set to "PDF") and "Language" (set to "English"). A red "Add to cart" button is located to the right of the language selection. At the bottom of the page, a link "Convert Swiss francs (CHF) to your currency" is visible.

## Abstract

This document defines data structures and basic operations on property graphs. It provides capabilities for creating, accessing, querying, maintaining, and controlling property graphs and the data they comprise.

This document specifies the syntax and semantics of a data management language for specifying and modifying the structure of property graphs and collections thereof.



# **How is it, making a standard?**

# Not like this!!!

It is not a musical request programme  
in a petting zoo  
next to the ice cream parlor  
in the land of milk and honey



# ... instead it is more like this

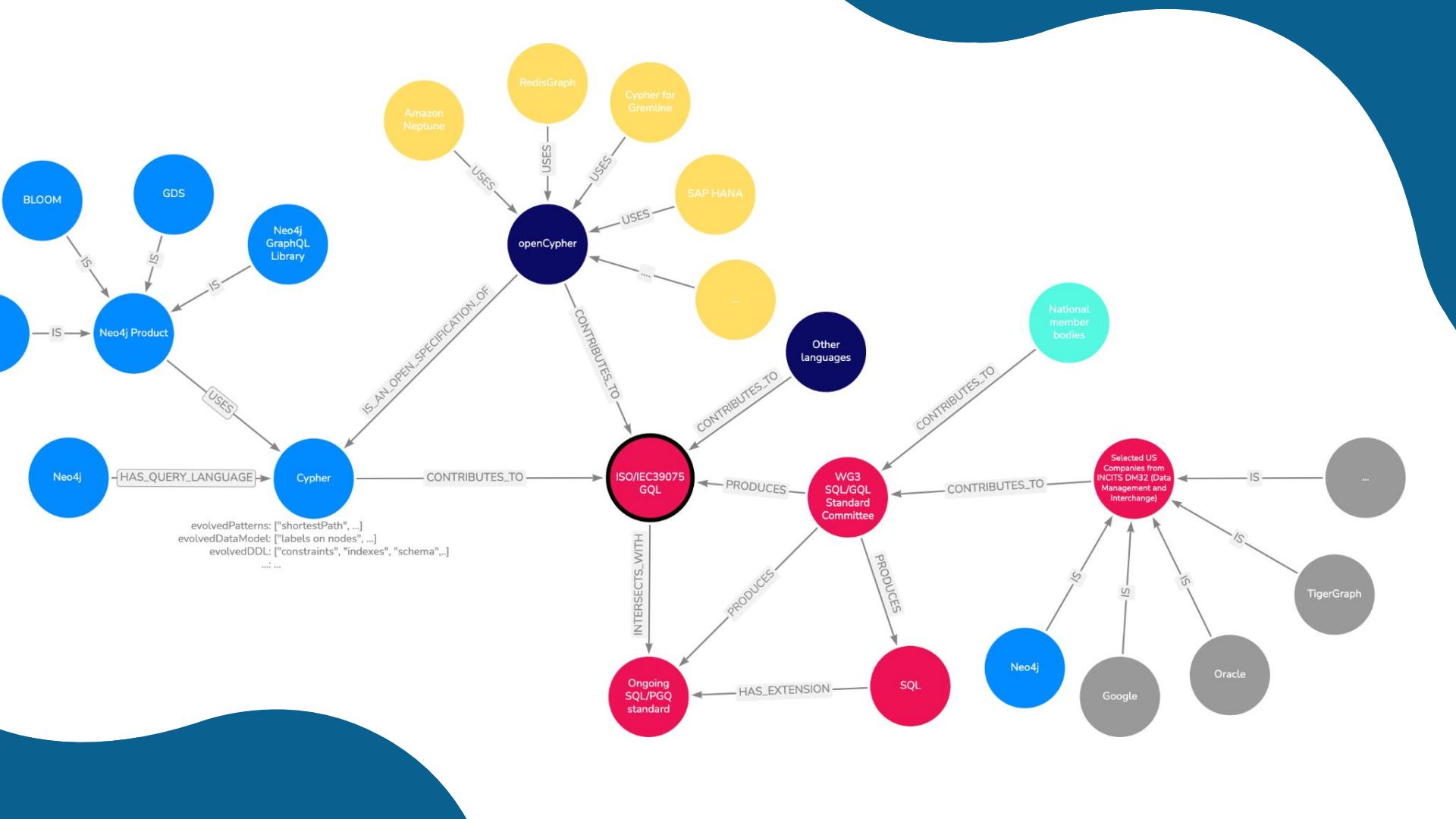


# The (non-technical) backdrop

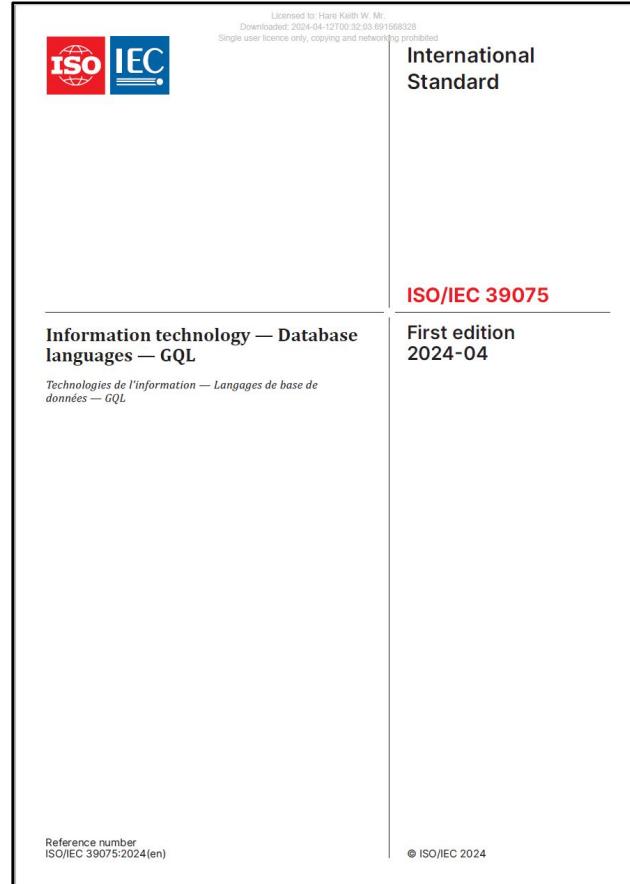
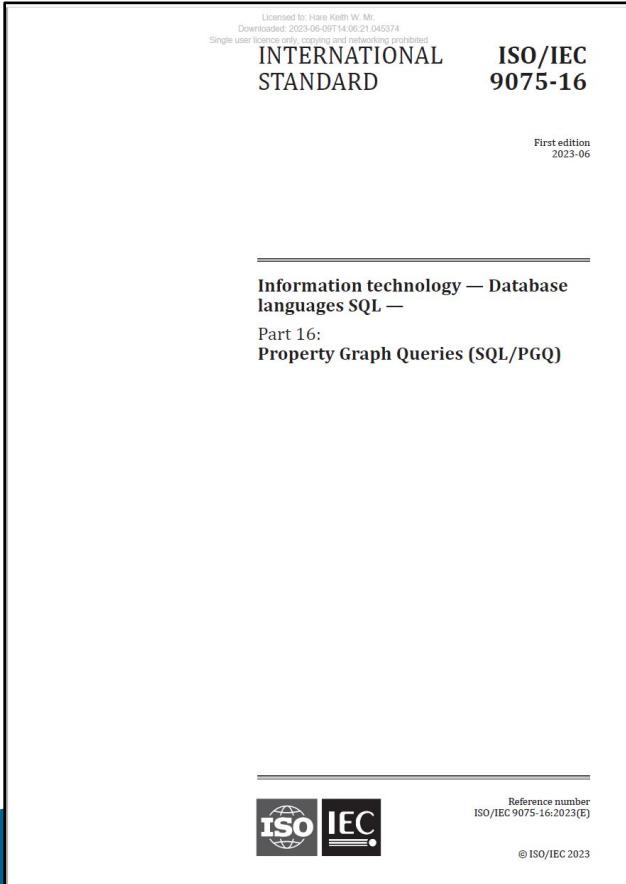
- Participants and actors are primarily database vendors
  - Who pay the show
  - Have stakes in the game (existing products, existing customers, market perspectives, matras, ambitions, etc.)
  - Have not necessarily compatible view points
  - Face significant negative business implications when breaking their things
- 30+ years matured process that
  - Enables reaching consensus among participants
  - Is proven to be capable of producing highly complex international standards
  - Requires to put in real work (go to meetings, write papers, run the marathon)
  - Caters for stability and reliability rather than the latest fancy idea
  - Comes with its own idiosyncrasies

# Not in a vacuum





# Property Graph Standards – SQL/PGQ and GQL



# Property Graph Standards – SQL/PGQ and GQL

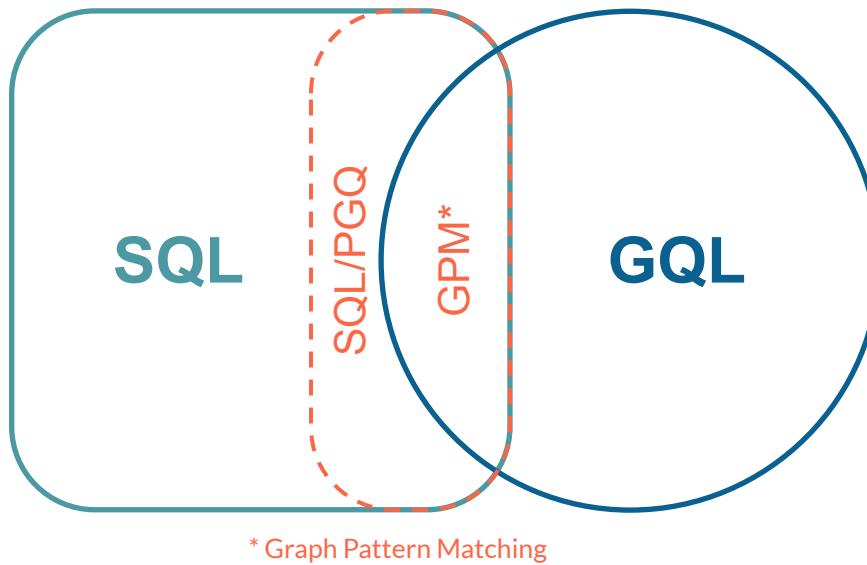
## SQL/PGQ

- Property Graph views of SQL tables
- **Graph Pattern Matching queries**
- GRAPH\_TABLE() in SQL FROM
- Supports Reads
- Common foundation with SQL and graph query languages
- Does not support schema-flexible graphs

## GQL

- Full DB language
  - DML – Create, Read, Update, Delete
  - DDL – Create Type, Create Graph
- **Graph Pattern Matching queries**
- Leverages common foundation from SQL and property graph languages
- Supports schema-fixed and schema-flexible variants

# Property Graph Standards – SQL/PGQ and GQL



# The SQL spec writing style

The image shows two pages from the ISO SQL standard document, IWD 9075-2:20xx(en), page 6.30 and 20.21.

**IWD 9075-2:20xx(en)**  
6.30 <numeric value expression>

### 6.30 <numeric value expression>

#### Function

Specify a numeric value.

#### Format

```
<numeric value expression> ::=  
  <term>  
 | <numeric value expression> <plus sign> <term>  
 | <numeric value expression> <minus sign> <term>  
  
<term> ::=  
  <factor>  
 | <term> <asterisk> <factor>  
 | <term> <solidus> <factor>  
  
<factor> ::=  
  [ <sign> ] <numeric primary>  
  
<numeric primary> ::=  
  <value expression primary>  
 | <numeric value function>
```

#### Syntax Rules

- Case:
  - If the declared type of either operand of a dyadic arithmetic operator is the decimal floating-point type, then the declared type of the result is the decimal floating-point type with an implementation-defined (IV134) precision.
  - If the declared type of either operand of a dyadic arithmetic operator is approximate numeric, then the declared type of the result is an implementation-defined (IV135) approximate numeric type.
  - Otherwise, the declared type of both operands of a dyadic arithmetic operator is exact numeric, and the declared type of the result is an implementation-defined (IV136) exact numeric type, with precision and scale determined as follows:
    - Let  $S1$  and  $S2$  be the scale of the first and second operands respectively.
    - The precision of the result of addition and subtraction is implementation-defined (IV136), and the scale is the maximum of  $S1$  and  $S2$ .
    - The precision of the result of multiplication is implementation-defined (IV136), and the scale is  $S1 + S2$ .
    - The precision and scale of the result of division are implementation-defined (IV136).
- The declared type of a <factor> is that of the immediately contained <numeric primary>.
- The declared type of a <numeric primary> shall be numeric.
- If a <numeric value expression> immediately contains a <minus sign>  $NMS$  and immediately contains a <term> that is a <factor> that immediately contains a <sign> that is a <minus sign>  $FMS$ , then there shall be a <separator> between  $NMS$  and  $FMS$ .

304

**IWD 9075-2:20xx(en)**  
20.21 <numeric value expression>

### 20.21 <numeric value expression>

#### Function

Specify a numeric value.

#### Format

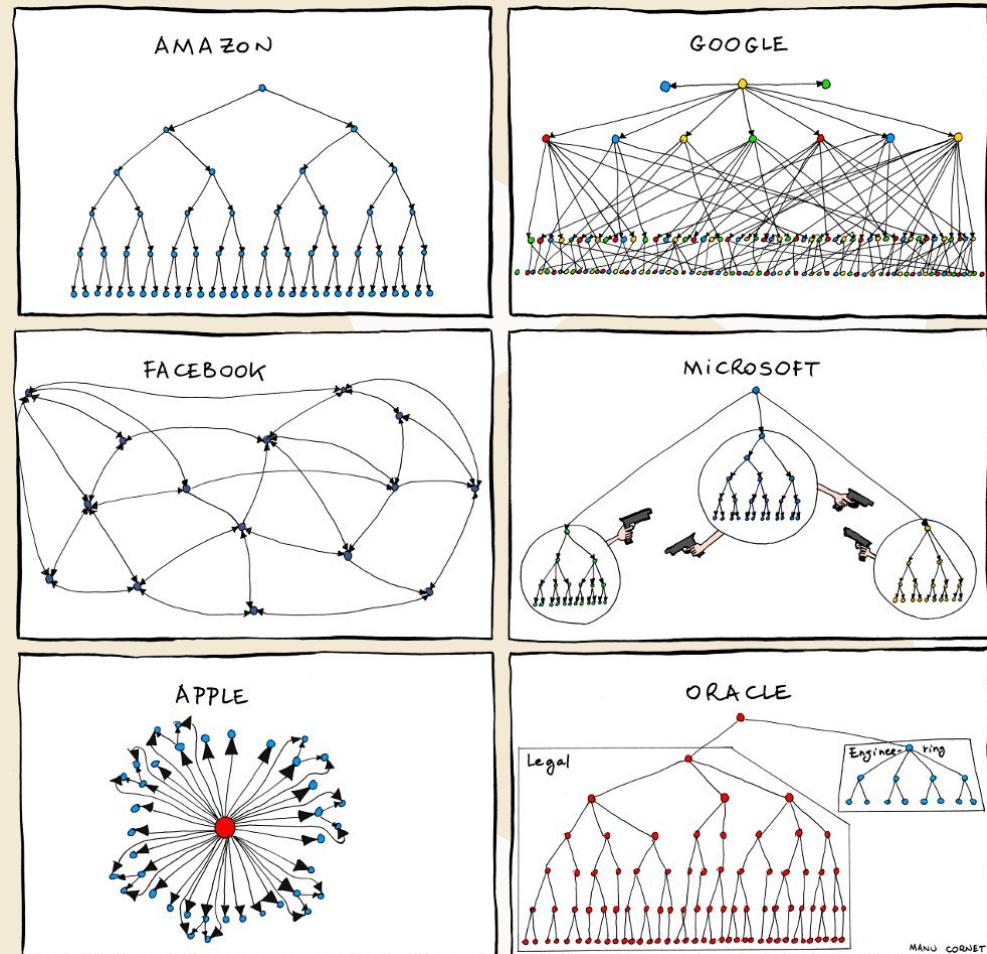
```
<numeric value expression> ::=  
  <term>  
 | <numeric value expression> <plus sign> <term>  
 | <numeric value expression> <minus sign> <term>  
  
<term> ::=  
  <factor>  
 | <term> <asterisk> <factor>  
 | <term> <solidus> <factor>  
  
<factor> ::=  
  [ <sign> ] <numeric primary>  
  
<numeric primary> ::=  
  <value expression primary>  
 | <numeric value function>
```

#### Syntax Rules

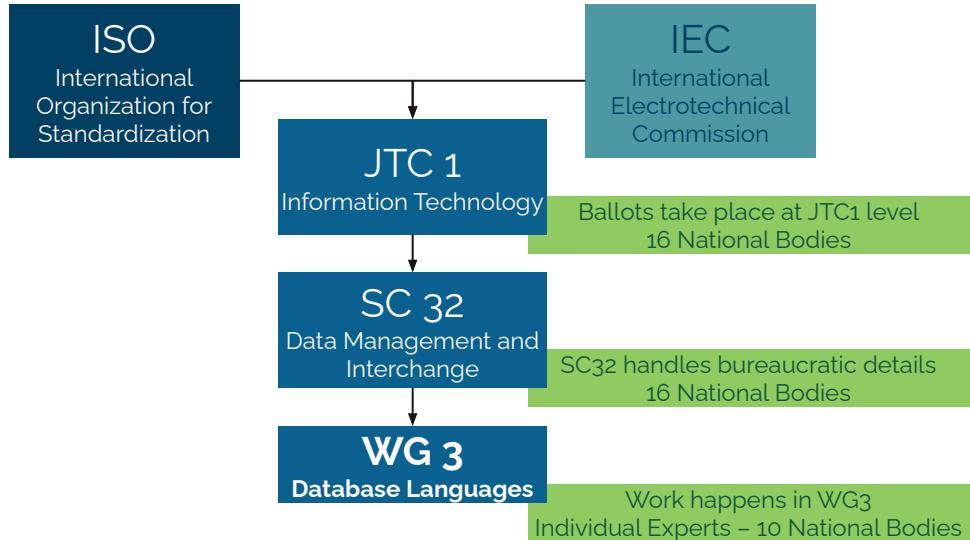
- Case:
  - If the declared type of either operand of a dyadic arithmetic operator is approximate numeric, then the declared type of the result is an implementation-defined (ID063) approximate numeric type.
  - Otherwise, the declared type of both operands of a dyadic arithmetic operator is exact numeric, and the declared type of the result is an implementation-defined (ID064) exact numeric type, with precision and scale defined as follows:
    - Let  $S1$  and  $S2$  be the scale of the first and second operands respectively.
    - The precision of the result of addition and subtraction is implementation-defined (ID065), and the scale is the maximum of  $S1$  and  $S2$ .
    - The precision of the result of multiplication is implementation-defined (ID066), and the scale is  $S1 + S2$ .
    - The precision and scale of the result of division are implementation-defined (ID067).
- The declared type of a <factor> is that of the immediately contained <numeric primary>.
- The declared type of a <numeric primary> shall be numeric.
- If a <numeric value expression> immediately contains a <minus sign>  $NMS$  and immediately contains a <term> that is a <factor> that immediately contains a <sign> that is a <minus sign>  $FMS$ , then there shall be a <separator> between  $NMS$  and  $FMS$ .

392

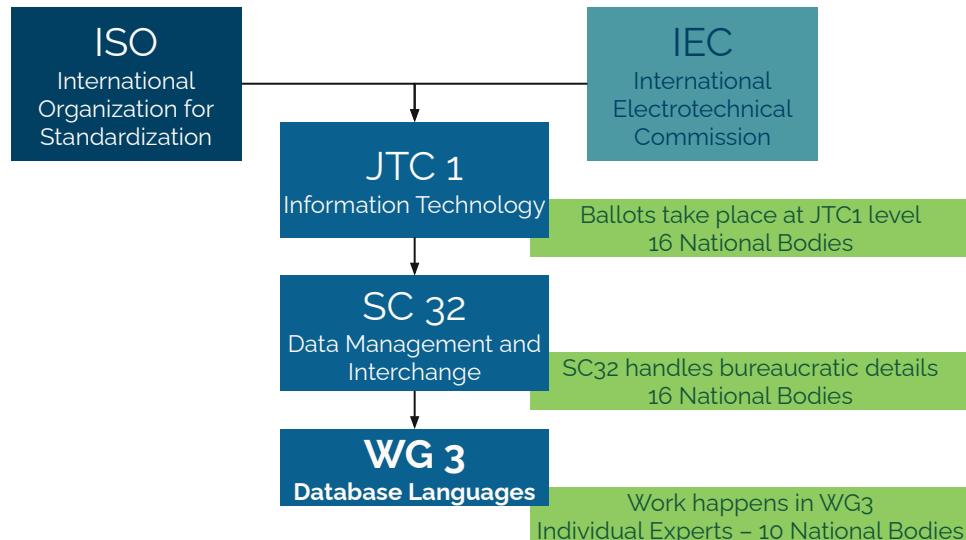
# The org charts



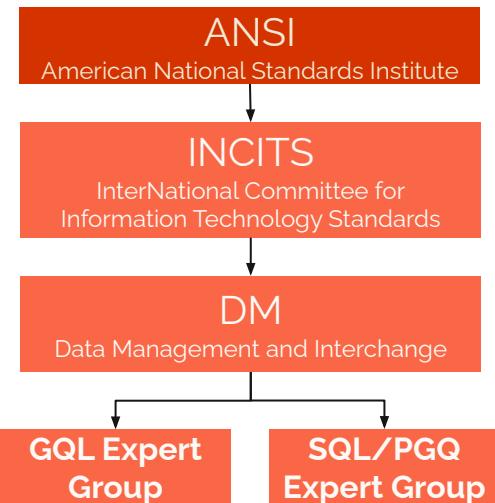
# International Standards Hierarchy



# International Standards Hierarchy



## US mirror



# Working in ...

## ISO/IEC JTC1 SC32 WG3

- Meetings
  - Week-long meetings two to three times a year – now with remote access
  - Monthly web conferences – three 3-hour sessions
- International group of national standards bodies
- Participants operate as individual experts
- Concrete change proposals
- Final decisions are made in WG3
- In practice much work happens within US Expert Groups

## INCITS Data Management

- Work done by interested parties in the Expert Groups
  - Property Graph Queries in SQL
  - GQL
- Expert Groups have weekly 2 hour calls
- Discussions in the Expert Groups is based on written proposals
- Proposals are either
  - Concrete change proposals
  - Discussion papers
- Ballot responses approved in INCITS Data Management

# Who participates – SC32 WG3?

Experts from the following 10 national bodies participate in SC32 WG3:

	China	→ Ant Financial, Boray Data, CESI, Huawei
	Denmark	→ TF Informatik
	Finland	→ Profium
	Germany	→ EDB, Oracle
	Japan	→ Hitachi, Tokyo Metropolitan University
	South Korea	→ Bundang Hospital, CnTechSystems
	Netherlands	→ Cannan Consultancy, EDB
	Sweden	→ Neo4j
	United Kingdom	→ PR Brown, University of Edinburgh
	United States	→ (see later slide)

# Who participates – INCITS DM32?

DM

Data Management and Interchange

## Mostly GQL

- ArangoDB Inc
- FairCom USA
- Google
- JCC Consulting Inc
- Neo4j Inc
- RelationalAI
- TigerGraph
- VESoft

## Mostly SQL

- Actian Corporation
- IBM Corporation
- Intersystems Corporation
- Microsoft Corporation
- Oracle
- SAP

## Mostly Metadata

- Farance Inc
- William McCarthy
- National Cancer Institute
- Nurocor

## Mostly Data Usage

- Department of Commerce – NIST

## Mostly Streaming SQL

- Boray Data

Note: The preliminary work on Streaming SQL has fizzled for the moment and a number of organizations that were primarily interested in streaming data have dropped out.

GQL Expert Group

SQL/PGQ Expert Group

# SC32 WG3 Formal Liaison Relationships

LDBC (Linked Data Benchmark Council) — liaison since 2017

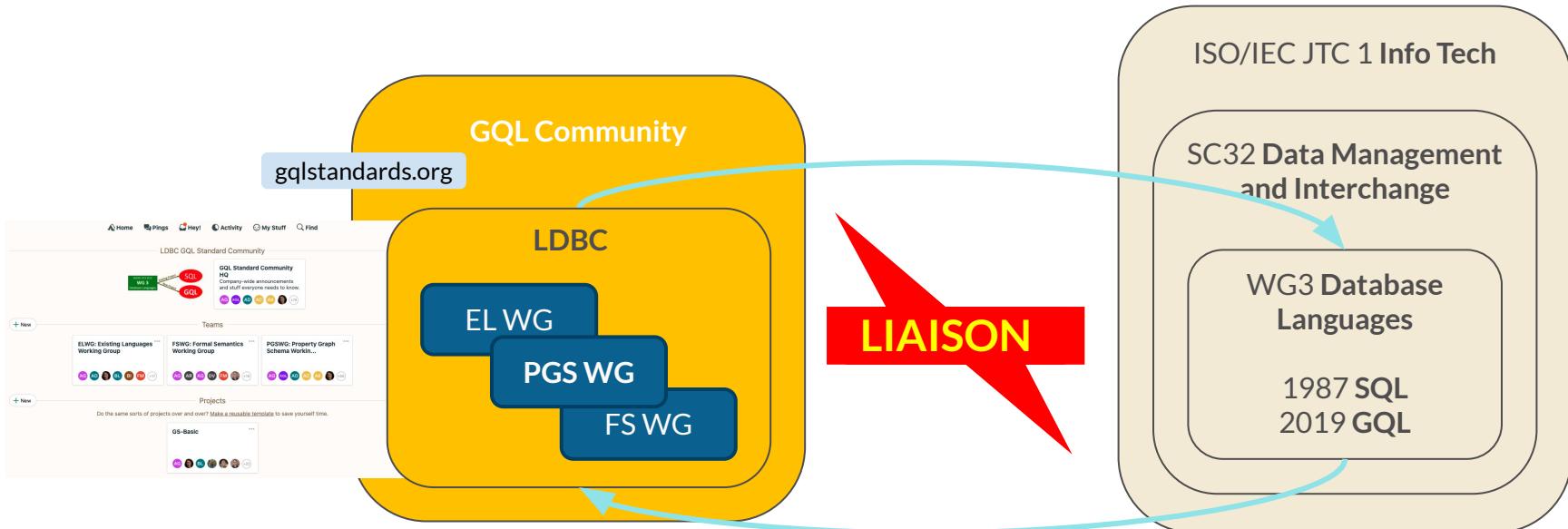
- Industry/Academic consortium focused on graphs
- Started with benchmarks & has evolved to model, language, and other topics
- Working Groups of interest to SC32 WG3 focus on property graph language—PGQ & SQL
- Existing Languages, Property Graph Schema, GQL Formal Semantics working groups
- Support/strengthen WG3 standards
- Review of WG3 documents
- Contribution of papers to WG3 (critique/corrections, feature suggestions)
- ACM SIGMOD/PODS 2021 paper “PG-Keys: Keys for Property Graphs”
- ACM SIGMOD/PODS 2023 best industry paper “PG-Schema: Schemas for Property Graphs”
- An evolving bi-directional process for collaboration

OGC (Open Geospatial Consortium)

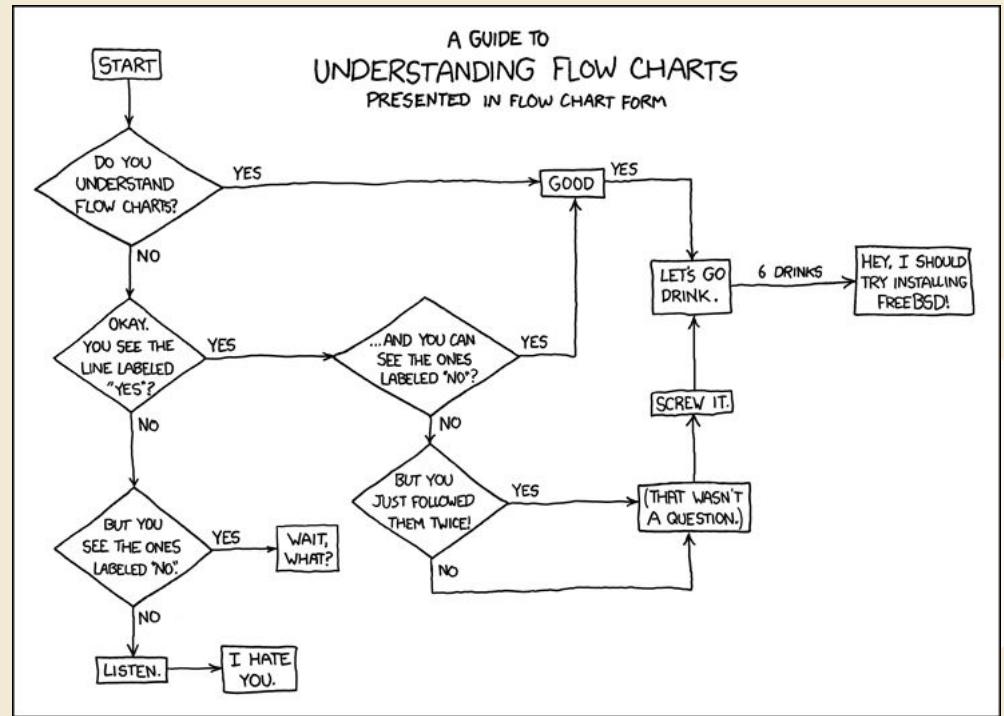
- Requirements for supporting spatial data in GQL (v2 or later)



# GQL community work

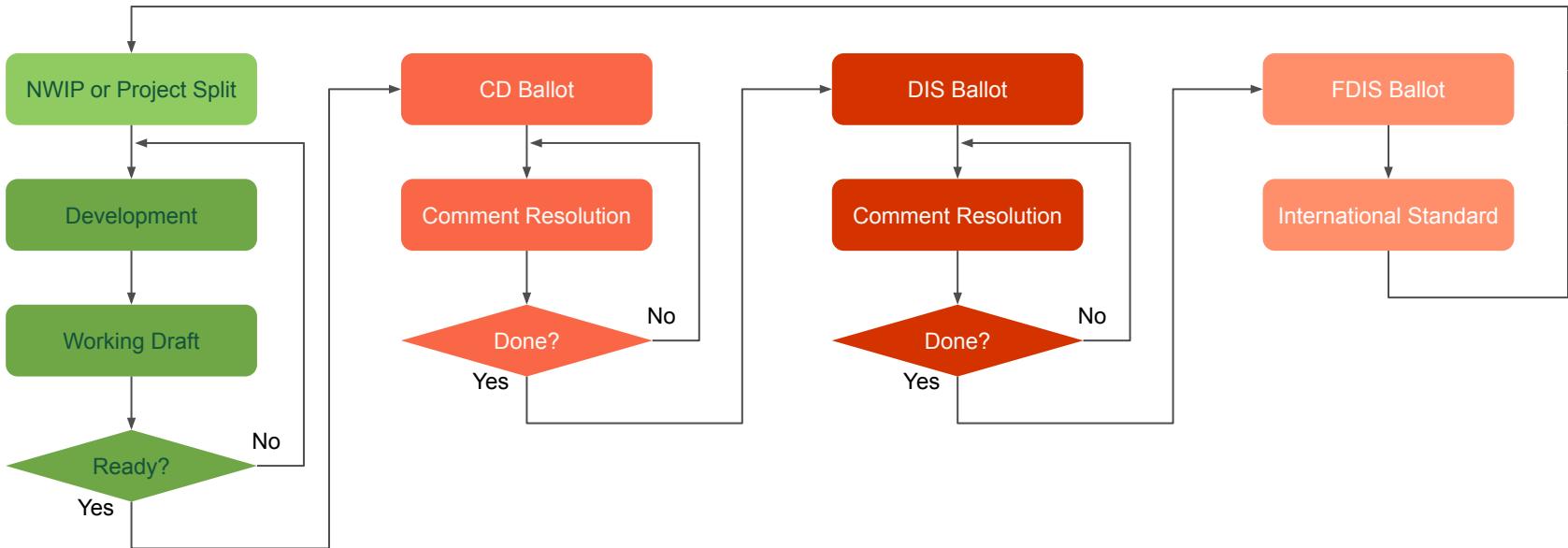


# Process



<https://xkcd.com/518/>

# ISO/IEC JTC1 Standardization Process





# Ballot comment resolution

## Label expression issue

TO RESOLVE BALLOT  
COMMENTS, YOU WRITE  
A PAPER – A CHANGE  
PROPOSAL.

**Author** Neo4j Query Languages Standards and Research Team\*

**Status** Change Proposal for [[GQL-1IWD32](#)]

<b>Revision</b>	<b>Date</b>	ISO/IEC JTC1/ SC32/WG3	INCITS DM32	DM32 GQL Expert Group
Original	2023-09-25	CMN-075	DM32-2023-00387	gql-2023-153
Revision 1	2023-09-25	CMN-075r1	DM32-2023-00387r1	gql-2023-153r1

- *Correctly referring to the graph type only when there is one*

## Abstract

This paper proposes a resolution to comment #306. P00-USA-103 on <label expression>.

# Ballot comment resolution

P00-USA-103		4-Minor Editorial	<i>P00-16.12, &lt;label expression&gt;</i>	SR 1) states the following: "Let <i>LE</i> be the <label expression> and let <i>GP</i> be the <graph pattern> that simply contains <i>LE</i> ."  In addition to this rule, SR 4) requires the label expression to be contained in a node pattern or an edge pattern.  But label expressions can instead be contained in a <labeled predicate>, or a <labeled predicate part 2> that is contained in a <simple when clause>. These rules should be amended to take into account of these cases.
<b>Solution</b> None provided with comment.				THE PAPER DISCUSSES THE COMMENT ...

The comment is correct in its observation that a <label expression> can be simply contained in non-terminal instances other than a <graph pattern> *GP*.

Specifically, a <label expression> can be simply contained in a <labeled predicate>.

The case where a <label expression> is simply contained in a <labeled predicate part 2> that is simply contained in a <simple when clause> is transformed by the Syntax Rules in Subclause 20.22, "<case expression>" to a <case expression> where the <label expression> is simply contained in a <labeled predicate>. So, we can focus on <labeled predicate>.

The comment points out that SR 1) and SR 4) need adjustment. The existing rules in Subclause 16.12, "<label expression>" are:

## « Syntax Rules

- 1) Let *LE* be the <label expression> and let *GP* be the <graph pattern> that simply

# Ballot comment resolution

## 2.2 Subclause 16.12, “<label expression>”

MODIFY SUBCLAUSE AS FOLLOWS

**Syntax Rules** *[Pale changes by [CMN-042]]*

1) Let *LE* be the <label expression> and let *GP* be the <graph pattern> that simply contains *LE*.

*[As in [CMN-045]]*

2) The current working graph site of *LE* shall be defined.

23) Let *PG CWGS* be the current working graph available at site of *LE*.

~~NOTE 195 — If no current working graph is available at *LE*, then this rule cannot be satisfied. See Subclause 4.7.3, “Working objects”.~~

3) Every <label name> contained in *LE* shall identify a label of *PG*. *[As in [CMN-045]]*

44) Case:

a) If *LE* is simply contained in an <is label expression>, then:

i) Let *GRVT* be the graph reference value type that is the declared type of *CWGS*.

ii) If *GRVT* is a closed graph reference value type with constraining GQL-object type *COT*, then:

i) If *LE* is simply contained in a <node pattern>, then *LE* is a node <label

... AND PROPOSES  
CHANGES TO THE DRAFT  
(OR TO CLOSE WITH NO  
ACTION)

# Work on the GQL draft between ballots

## GQL Expert Group\*

The screenshot shows a web-based application for managing GQL proposals. At the top, there's a header with the title 'GQL Expert Group\*' and some navigation links. Below the header, there's a section titled 'Performance Rules' with a note about changes to subclause 15.3. The main area displays a list of proposals, each with a thumbnail icon, a title, and a small preview of its content. A large green arrow points from this section to the 'WG 3 Database Languages' section.

Experts produce  
and agree to  
specific changes  
to the draft in  
form of change  
proposal  
documents

\* Or experts in other national bodies  
or a liaison

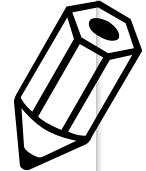
## WG 3 Database Languages

Experts debate  
and agree to  
change proposal  
documents

- Title w16032R2\_Correct-maximi
- w16033\_CD\_9075-16\_SQL
- w16034-response-to-W15-I
- w16034r2-response-to-W1!
- w16035-Referential\_action
- w16036-Defect\_Report\_#11
- w16037\_Simple\_Graph\_Patt
- w16038\_Convert\_Create\_and
- w16039\_Variable\_names\_an
- w16040\_Remove\_two\_kinds
- w16041\_Convert\_match\_graj
- w16042\_Response-to-W16-
- w16043\_nonterminating-su

The screenshot shows a web-based application for managing GQL proposals. At the top, there's a header with the title 'WG3:W15-008' and some metadata. The main area displays a list of proposals, each with a thumbnail icon, a title, and a small preview of its content. A large green arrow points from the 'WG 3 Database Languages' section to the 'Editors# modify the draft accordingly' section.

Editors# modify the  
draft accordingly



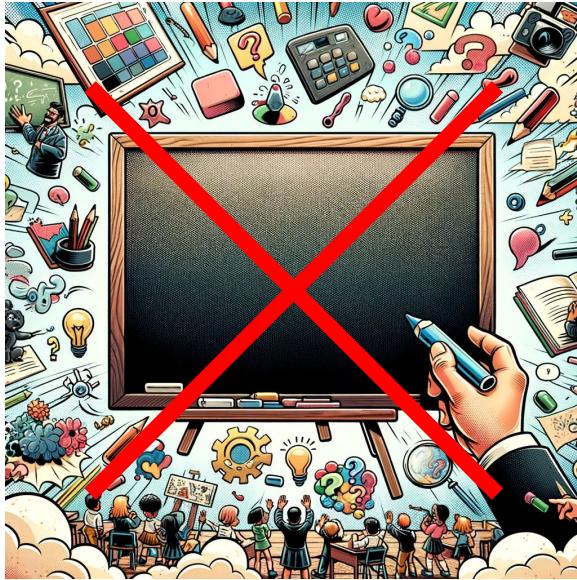
# Stefan Plantikow and Stephen Cannan

# Work in numbers

- The GQL standard was published on April 11 or 12, 2024 (depending on your timezone)
- The GQL standard is **628 pages**, about the same size as the SQL:92 standard.
- **38 WG3 meetings** to produce the GQL standard (11 face-to-face, 27 online).
- Source XML for the GQL standard references **~430 papers**  
with a **total of over 7700 pages** and an **average of ~18 pages** per paper  
from **~20 authors**.
- In the February 2023 5-day meeting, we reviewed and accepted 85 GQL papers.
- The longest change proposal was a total of 177 pages, although the last 100 pages were examples illustrating the results of the proposed changes.

# Takeaways

Not a blank slate



It is a ton of work



# Thank you!

[hannes.voigt@neo4j.com](mailto:hannes.voigt@neo4j.com)