SupplementaryFile 2 (S2) - Methods-details

S2.1. Brief overview on the graph database (GDB) technologies and associated publications

In Table S2.1, we provide the details on the data model, initial release, licence type, and the number of associated publications from PubMed (pubmed.ncbi.nlm.nih.gov) or PubMed Central (PMC; ncbi.nlm.nih.gov/pmc) for the top GDB technologies (both open-source and commercial) as reported in the DB-Engines resource (reference date 09/2023).

Table S2.2 offers a comparison of the four popular open-source GDBs: Neo4, ArangoDB, OrientDB, and Virtuoso. Currently, Neo4j is the most widely used open-source GDB tool; however, there is also a comparable number of works that use the Virtuoso, ArangoDB, and OrientDB multi-model databases (DB), which combine different types of non-relational DBs simultaneously, for example allowing the use of document-oriented data to include queries through publications linking them to the biological objects they describe.

Table S2.1. Ranking of the top 16 graph open-source and commercial DBs based on DB-Engines (db-engines.com, reference date 09/2023), an initiative to collect and present information on database management systems. We include the number of articles found in PMC that use or mention these databases [reference date for PMC check: 09/2023].

#	Graph database name	Database model	Initial Release	Licence	PMC*	Rank
1	Neo4j**	Graph	2007	Community Edition: GPLv3	544	50.39
2	Microsoft Azure Cosmos DB	Multi-model	2014	Commercial	1	35.45
3	Virtuoso**	Multi-model	1998	Open Source Edition: GPLv2	69	5.38
4	OrientDB**	Multi-model	2010	Community Edition: Apache 2	35	4.33
5	ArangoDB**	Multi-model	2012	Free Edition: Apache 2	25	4.29
6	Memgraph	Graph	2017	Commercial	1	2.88
7	GraphDB	Multi-model	2000	Commercial	18	2.6
8	Amazon Neptune	Multi-model	2017	Commercial	2	2.54
9	JanusGraph***	Graph	2017	Apache 2	7	2.39
10	Nebula Graph***	Graph	2019	Apache 2	141	2.33
11	Stardog	Multi-model	2010	Commercial	6	2.28
12	TigerGraph	Graph	2017	Commercial	5	2.21
13	Dgraph***	Graph	2016	Apache 2	6	1.89
14	Fauna	Multi-model	2014	Commercial	4	1.69
15	Giraph***	Graph	2013	Apache 2	4	1.65
16	AllegroGraph**	Multi-model	2013	Commercial; Free edition	36	1.15

*This column is based on authors' analysis for the number of hits in PMC publications, last update 09/2023; **Commercial with open source or free version available; ***Open source

Table S2.2. Comparison of open-source GDB based on the information in the DB-Engines initiative (db-engines.com): Neo4, ArangoDB, OrientDB and Virtuoso.

Database	Database type	Models included	Query language	Release year	Implement ation	Data Scheme	SQL support
Neo4j	Graph	Graph	Cypher	2007	Java	Schema-free and schema-optional	Yes*
ArangoDB	Multi-model	Document; Graph; Key-value; Search engine	AQL	2012	C++	Schema-free	No
OrientDB	Multi-model	Document; Graph; Key-value;	Gremlin	2010	Java	Schema-free	No
Virtuoso	Multi-model	Document; Graph; Native XML; Relational; RDF; Search engine	SPARQL	1998	C	SQL - Standard relational schema; RDF - Quad/Triple; XML - DTD, XML schema	Yes

*The Neo4j Enterprise distribution includes the BI Connector, a JDBC-compatible interface, that allows executing SQL queries over a Neo4j resource.

S2.2. PubMed and PMC search queries

PubMed (pubmed.ncbi.nlm.nih.gov) and PubMed Central (PMC; ncbi.nlm.nih.gov/pmc) were searched for relevant publications with a cut-off date of 31/03/2023 using the key words "graph database" or "graph databases", including the top 16 popular graph databases according to DB-Engines (as given in Table S2.1) such as Neo4j, Azure Cosmos, ArangoDB, etc. The key words used for the search were chosen in order to maximise the specificity of the results in relation to graph database publications (thus, to minimise the number of accidental matches). For example, for the Virtuoso graph database, it is "Openlink AND Virtuoso", since simply searching for "Virtuoso" returned instances where the key word was matched to the family name of authors in PubMed/PMC instead of the graph database technology. Other examples are "Apache AND Giraph" and "Fauna AND graph database". For Stardog, we used "Stardog AND graph database" because it is one of the multi-model DBs and is often used as an RDF store and not specifically as a GDB. For unique names we directly used the names of the DBs such as "Neo4j" or "Azure Cosmos DB" or "ArangoDB".

Specifically, we used Search Query #1, since the search in PubMed found key phrases in titles and abstracts, results containing only the abstract term “graph database” were also included, as they could have specific DBs used in the work, but not mentioned in the title. This search query covered most publications where GDBs were mentioned.

```
Search Query #1: "graph database" OR "graph databases" OR Neo4j OR "Azure Cosmos" OR ArangoDB OR OrientDB OR (Openlink AND Virtuoso) OR (Ontotext AND GraphDB)  OR JanusGraph  OR "Amazon Neptune" OR (Stardog AND "graph database") OR TigerGraph OR FaunaDB OR (Fauna AND "graph database") OR AllegroGraph  OR (Dgraph AND "graph database") OR (Giraph AND "graph database") OR "Nebula Graph" OR Memgraph
```

Then, we further used Search query #2 to search specifically for mentions of specific GDBs (including Neo4j, ArangoDB, etc.) in full-text publications in PMC in order to prioritise the publication list: a publication presenting a direct use of a specific GDB technology was given higher priority than a publication mentioning only the GDB technology.

```
Search Query #2: Neo4j OR "Azure Cosmos" OR ArangoDB OR OrientDB OR (Openlink AND Virtuoso) OR (Ontotext AND GraphDB)  OR JanusGraph  OR "Amazon Neptune" OR (Stardog AND "graph database") OR TigerGraph OR FaunaDB OR (Fauna AND "graph database") OR AllegroGraph  OR (Dgraph AND "graph database") OR (Giraph AND "graph database") OR "Nebula Graph" OR Memgraph
```

Search query #2, focusing on specific GDBs, acts as a refinement criteria for our analysis: for full-text PMC publications, we considered only those mentioning the top 16 GDB technologies/ approaches.

S2.3. Python script to create the results table

We developed a Python script to merge the results of these search queries via PMID, PMCID, and DOI, available in CSV files, thus consolidating a single table with all relevant information. The Python script and its output is available at github.com/ilyamazein/gdbreview.

S2.4. Manual review: criteria for inclusion/exclusion and subdividing into categories

After removing duplicated publications (n=146), we aggregated a list of n=681 publications to be screened for this review. Each shortlisted publication was manually annotated by two reviewers. Several important inclusion/ exclusion criteria followed during the manual review allowed accelerating the process, minimising time and effort for further review of a full-length publication:

1. First, we considered only publications with the full text accessible to us (open or via our institutes). We also considered only publications with the text provided in English.
2. Second, we checked and confirmed that a certain GDB technology was not simply mentioned but was actually applied in the work described. Publications that only mention but not use a GDB were removed. We also removed preprints or conference posters.

3. Third, we removed publications describing integrated resources that were not available at the location mentioned in the publication nor provided a repository for the source code.
4. We selected only publications where a GDB was used in bioinformatics or systems biology context. For the COVID-19 knowledge bases, we extended the inclusion to systems biomedicine as the COVID-19 resources themselves can be classified as interdisciplinary, incorporating medical and other data.

We grouped these according to their content in several major categories. Table S2.3 includes the respective definitions, the number of publications selected per category, and the section in this review addressing them in detail. Note: a few SOFTWARE publications were assigned to multiple categories given that details were provided also for the methodological approach. For example, one may be considered as a SOFTWARE and an ONTOLOGY publication given detailed description of both the integrated knowledge base as well as the network-based method used for the analysis.

Table S2.3. Details on the major categories of publications annotated during the review, based on their content: the definition, the number of selected publications per each category and the review section describing a specific publication category

Category name	**Definition**	**Name of the section in this review addressing details**
REVIEW	if the publication was a review dedicated to the application of GDB approaches in bioinformatics and systems biology;	Information from these reviews was integrated in the main text, when needed.
METHOD	if the publication presented a method developed using a GDB approach for addressing a specific problem or question in systems biology;	Mainly presented in the “Analytical approaches and tools enabled by GDBs” section.
SOFTWARE	if a tool developed using a GDB approach for research was described. In this case, the publications were refined based on the availability of the resource itself to be queried or of the development code (e.g. github repository), and only those providing supporting urls for these points were retained for full text consideration. Supplementary FIle Software includes details on the software/ tools regarding the name, availability, and updates;	Software tools were described in the main text, when needed.

PRIMARY	if the publication presented a GDB version of original resources. For example, Reactome graph database. These PRIMARY resources can be used for developing more complex resources that include multiple sources (see INTEGRATED below);	Mostly presented in the “Pathway and network exploration“ section, but the primary resources were described in the main text, when needed.
INTEGRATED	if the publication described an GDB resource developed by integrating multiple DBs from systems biology (e.g. resources on pathways, biomarkers and drug-targets);	This is the most extended category of publications included in the review. They were described in the main text, when needed.
ONTOLOGY	if a GDB approach to describe and facilitate access to the terms of an ontology was presented.	Mainly presented in the “Ontologies” section.
OTHERS	if a GDB approach was used for purposes different than above, such as integration of web semantic data or of medical data etc.	Publications marked as OTHERS were removed during the full text revision, being considered out of scope.

Further, we manually annotated each publication with details on the GDB technology used in the publication (e.g., Neo4j, Virtuoso, OrientDB), the name, and the url in the case of integrated resources/ softwares/ tools as well as maintenance information if available. We also briefly summarised the content of the publication, and extremely important, we annotated reasons on including/ excluding the respective publication in the current review.