



RAGs to reqs

Irene Michlin

AppSec Lead, Neo4j



Making ASVS more accessible through graphs and LLMs



Agenda

1. Architecture and concepts
2. Demo
3. How we use it
4. Ideas for future development



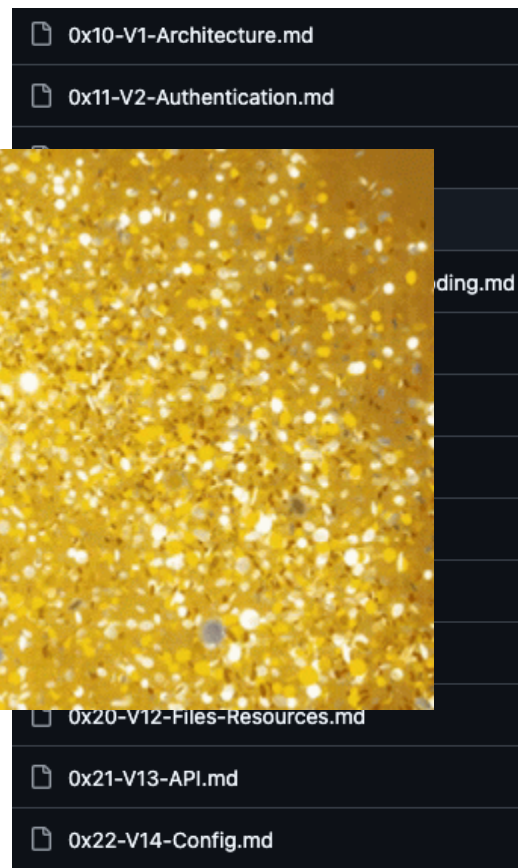
What is ASVS?

The OWASP Application Security Verification Standard (ASVS) Project provides a basis for testing web application technical security controls and also provides developers with a list of requirements for secure development.



Many ways to use it

- Secure Coding
- Security Archite
- Guide for auto
integration tes
- Secure Develop
- Framework for
secure softwar
- Agile Applicati



Example

V3.1 Fundamental Session Management Security

#	Description	L1	L2	L3	CWE	NIST §
3.1.1	Verify the application never reveals session tokens in URL parameters.	✓	✓	✓	598	

V3.2 Session Binding

#	Description	L1	L2	L3	CWE	NIST §
3.2.1	Verify the application generates a new session token on user authentication. (C6)	✓	✓	✓	384	7.1
3.2.2	Verify that session tokens possess at least 64 bits of entropy. (C6)	✓	✓	✓	331	7.1
3.2.3	Verify the application only stores session tokens in the browser using secure methods such as appropriately secured cookies (see section 3.4) or HTML 5 session storage.	✓	✓	✓	539	7.1
3.2.4	Verify that session tokens are generated using approved cryptographic algorithms. (C6)		✓	✓	331	7.1

Tailor it!

<https://www.infoq.com/presentations/sustainable-security-requirements-asvs/>



Lies, big lies, and LLMs

USER

Any specific ASVS requirements to keep in mind for this feature?

ASSISTANT

The Application Security Verification Standard (ASVS) from OWASP provides a

bro please
respond in valid json format
without errors and
make super sure the
syntax is extra correct
i'm begging you... and
please, pretty please,
don't make up answers
my career depends on it bro



**PROMPT
ENGINEER**

imgflip.com

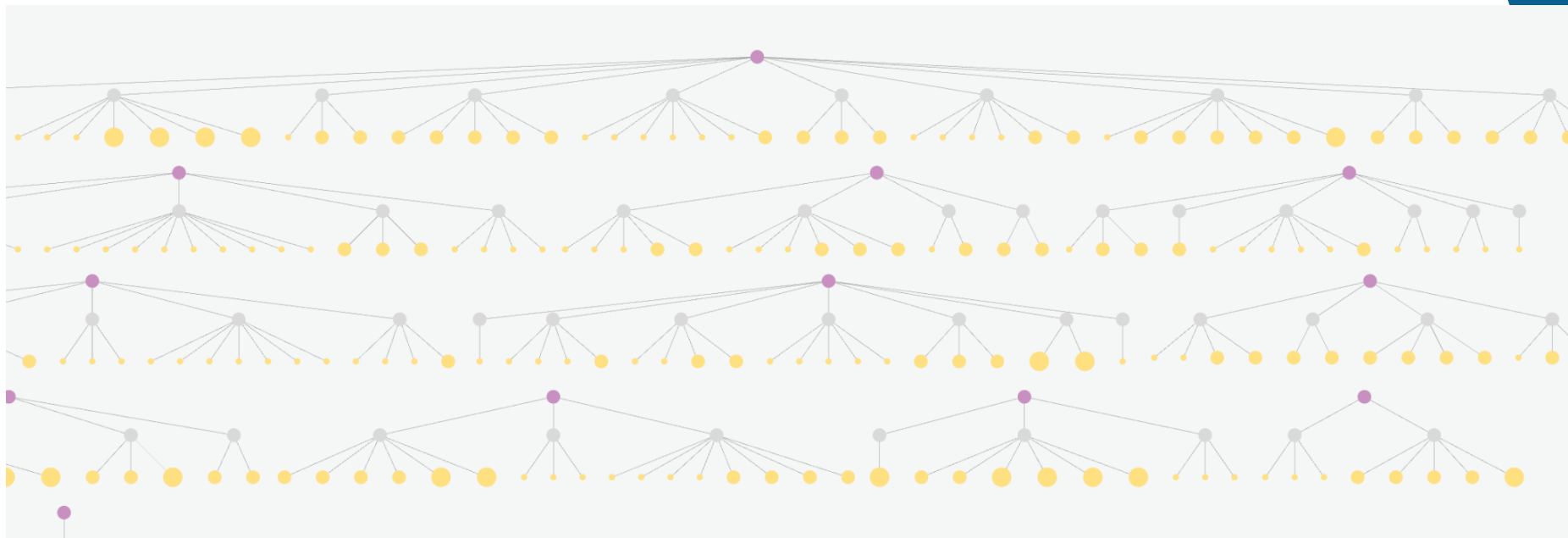


ASVS is “graphy”

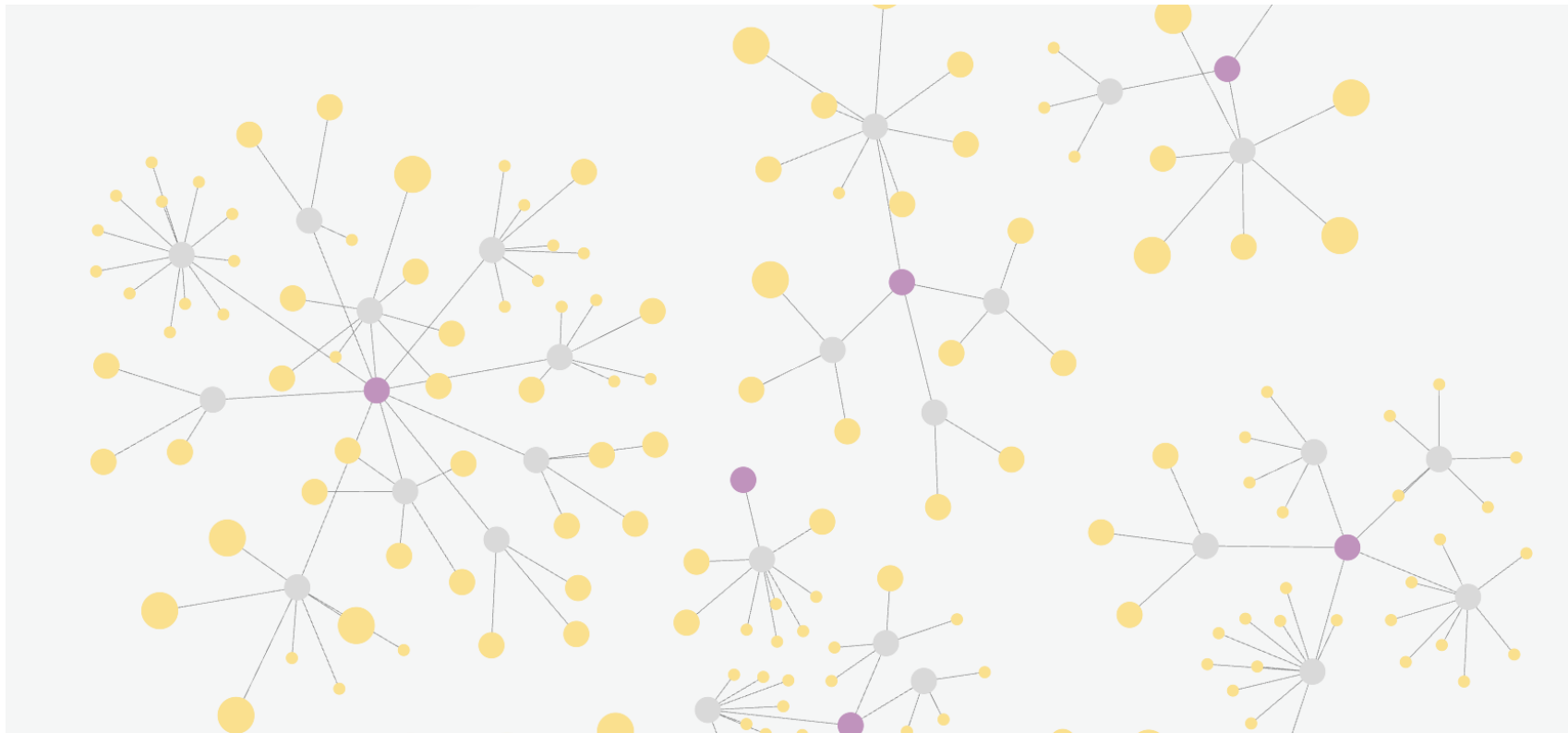
Chapter->Section->Requirement

V4 -> V4.1 -> V4.1.1

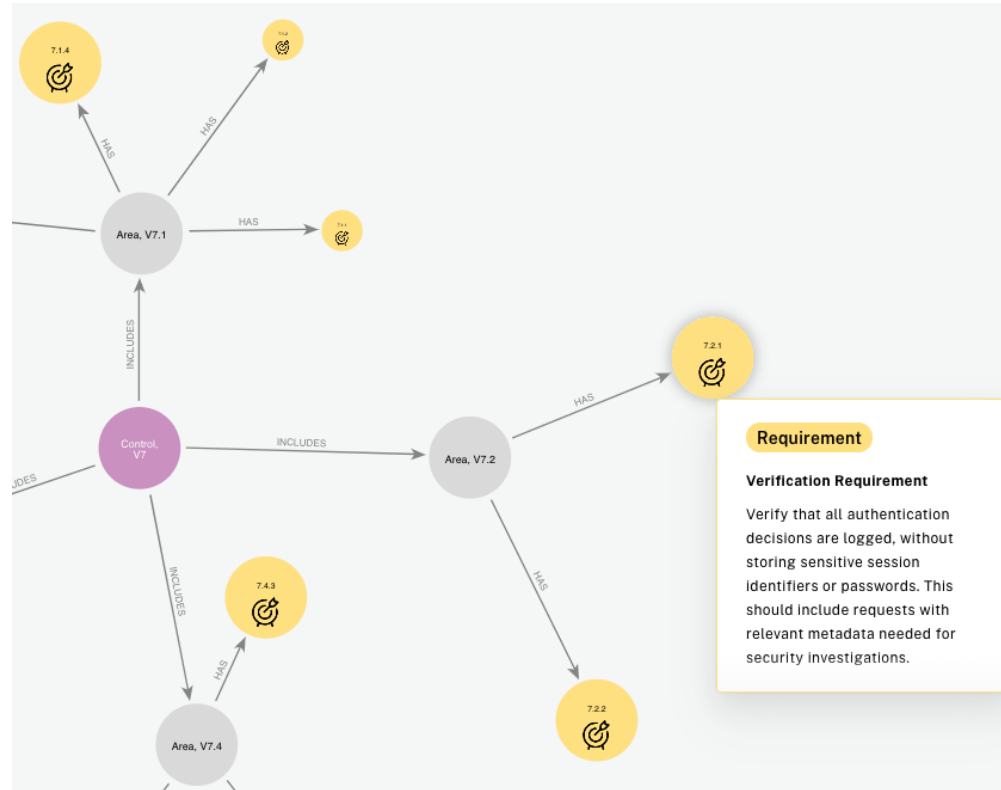
Hierarchical view



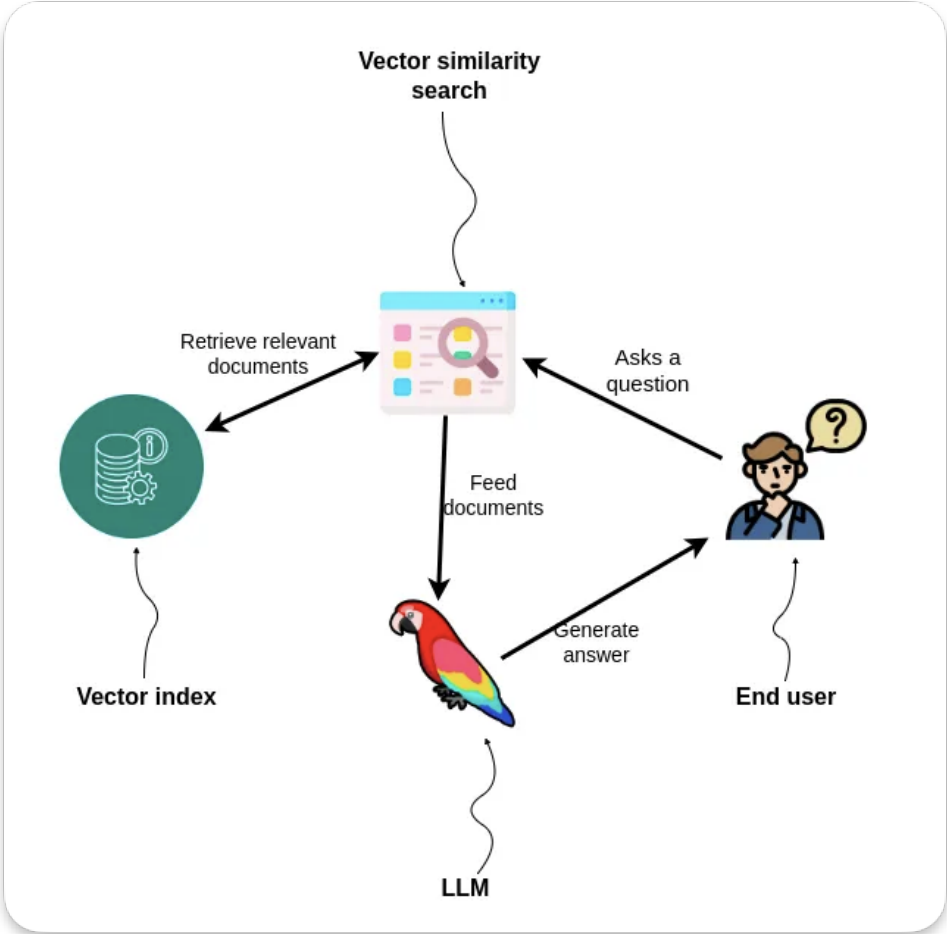
Force-based layout



Zoom-in



Architecture

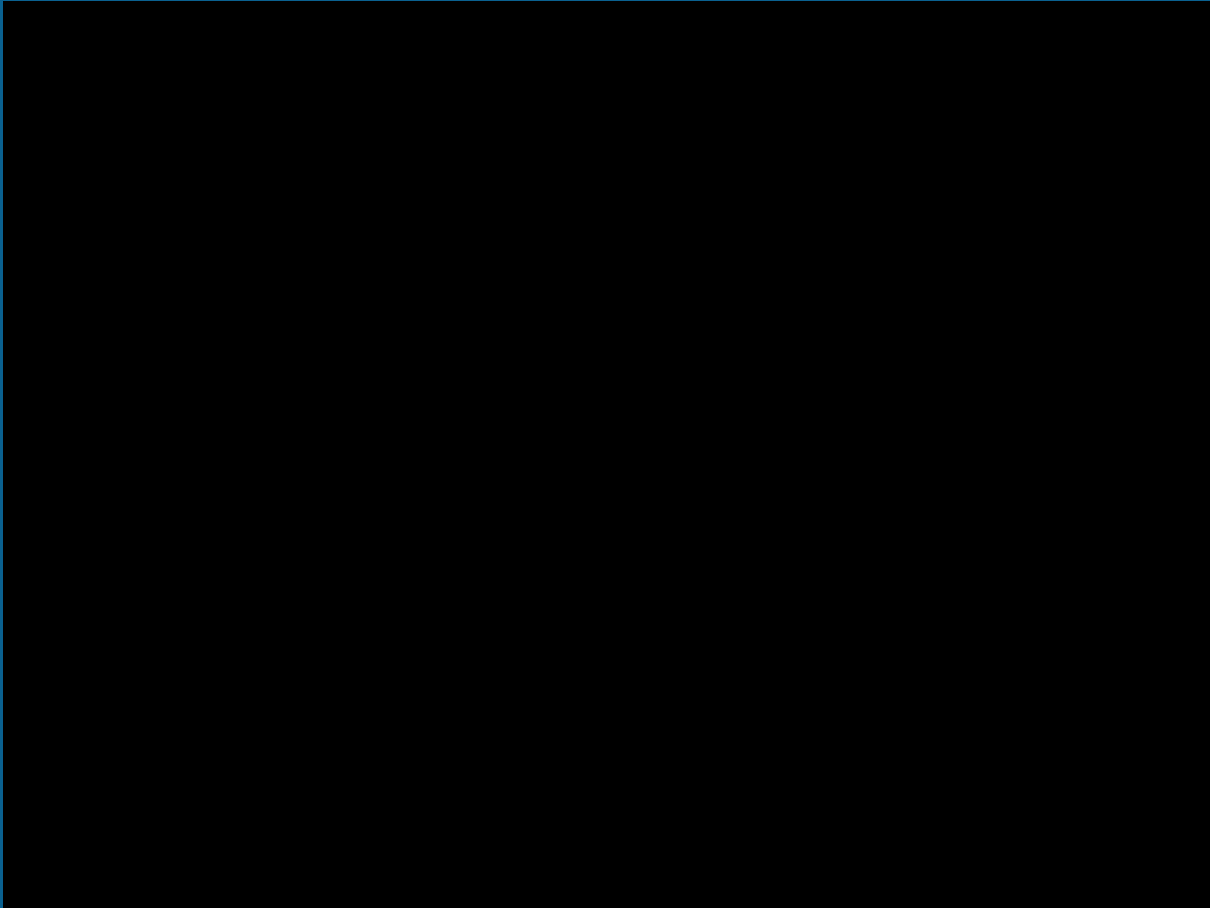


Demo

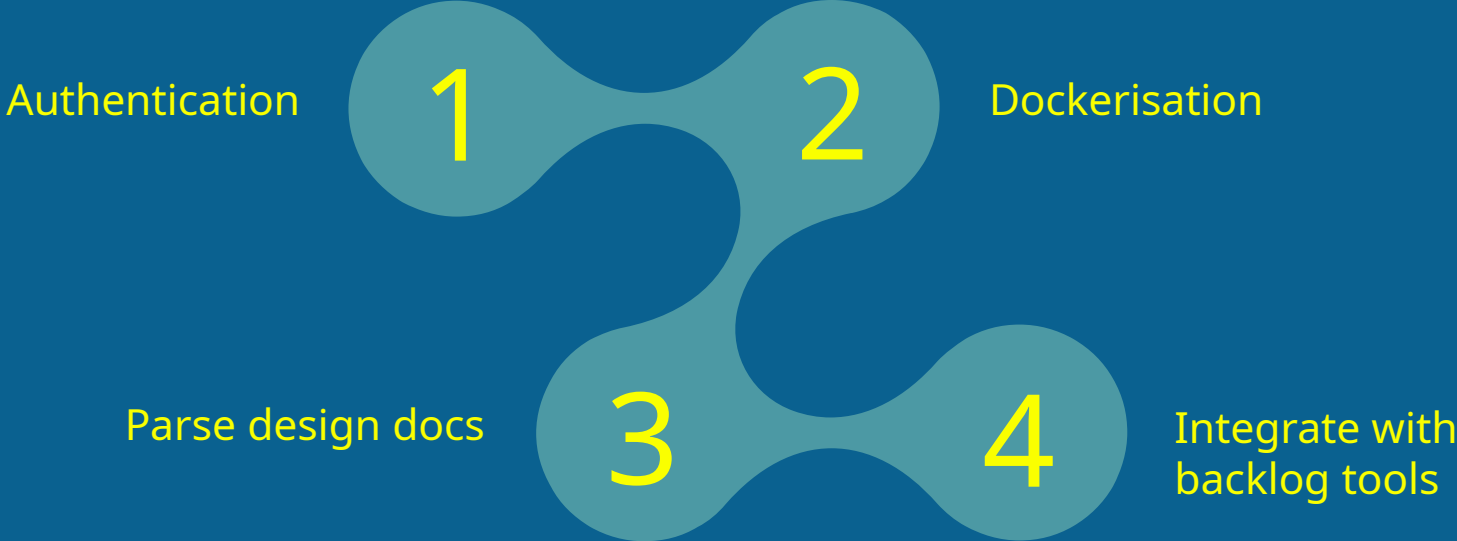
All the tools you need for this are free: <https://github.com/neo4j-examples/appsec-asvs-bot>

(except for OpenAI subscription)





Ideas for further development





How to update my graph?

V5 is coming

Maybe 4.0.4 is coming?



Use Data Importer

The screenshot displays the Neo4j Data Importer interface. On the left, a file list shows 'owasp_asvs.csv' with columns: chapter_id (V1), chapter_name (Architecture, Design and ...), section_id (V1.1), section_name (Secure Software Develop...), req_id (V1.1.1), req_description (Verify the use of a secure...), level1, level2 (✓), level3 (✓), cwe, and nist. The main workspace contains a large blue circle labeled 'Chapter'. On the right, the 'Definition' tab is active, showing 'Label' set to 'Chapter' and 'File' set to 'owasp_asvs.csv'. The 'Map from file' section shows a mapping of CSV columns to the 'Chapter' label, with 'chapter_id' and 'chapter_name' checked.

File	Chapter
chapter_id	V1
chapter_name	Architecture, Design and ...
section_id	V1.1
section_name	Secure Software Develop...
req_id	V1.1.1
req_description	Verify the use of a secure...
level1	
level2	✓
level3	✓
cwe	
nist	

<https://workspace-preview.neo4j.io/workspace/import>

Use LLM Knowledge Graph Builder

The screenshot displays the Neo4j Knowledge Graph Builder interface. On the left, there are four data source options: 'Drag & Drop' (Documents, Images, Unstructured text), 'Web Sources', 'Amazon S3', and 'GCS'. The main area shows a 'Neo4j connection' status of 'Not Connected' with a 'Graph Schema configured(4 Labels + 3 Rel Types)'. A 'Connect to Neo4j' modal is open, prompting the user to 'Drop your neo4j credentials file here' or 'browse'. The modal includes fields for Protocol (neo4j+s), URI (4d695339.databases.neo4j.io:7687), Database (neo4j), Username, and Password. A table with columns Name, Status, Upload Status, Size (KB), Source, and Type is visible in the background. A welcome message on the right states: 'Welcome to the Neo4j Knowledge Graph Chat. You can ask questions related to documents which have been completely processed. 06/11/2024 17:12:21'.

<https://llm-graph-builder.neo4j.com/>

Build it programmatically

neo4j-graphrag 1.2.0



Released: Oct 28, 2024

```
pip install neo4j-graphrag
```

Python package to allow easy integration to Neo4j's GraphRAG features

Navigation

Project description

Release history

Download files

Verified details

These details have been [verified by PyPI](#)

Project links

Project description

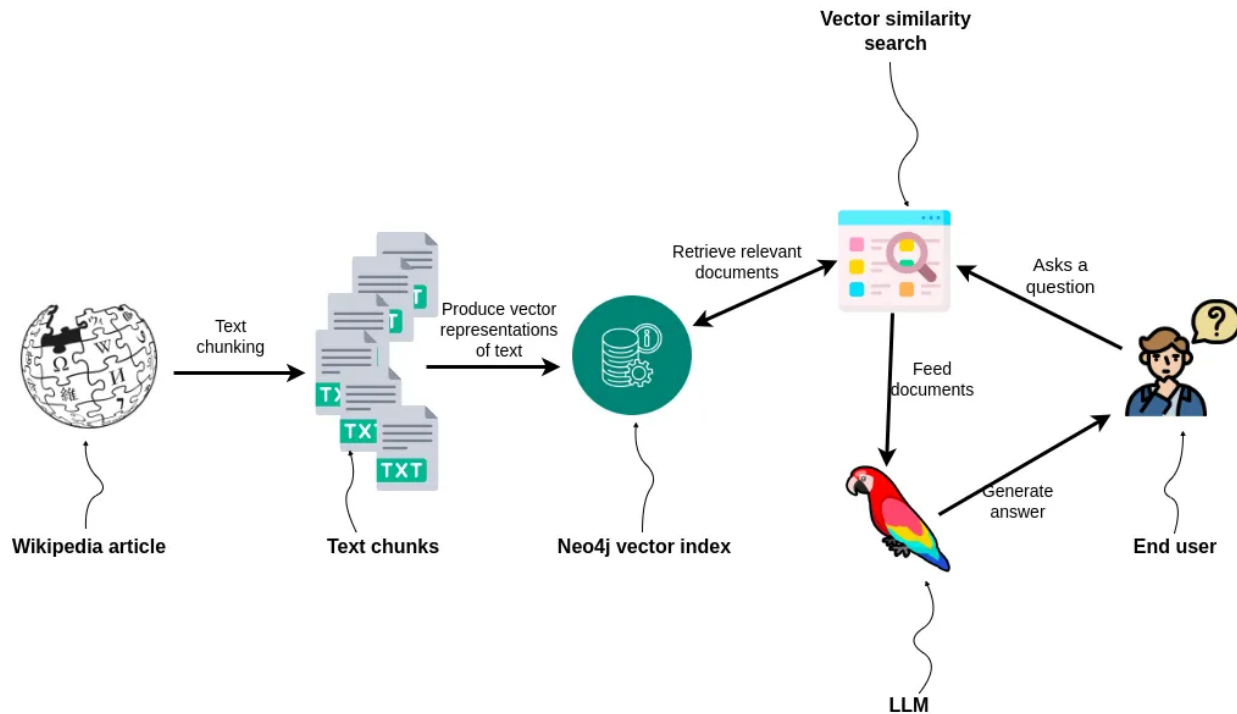
Neo4j GraphRAG Package for Python

The official Neo4j GraphRAG package for Python enables developers to build [graph retrieval augmented generation \(GraphRAG\)](#) applications using the power of Neo4j and Python. As a first-party library, it offers a robust, feature-rich, and high-performance solution, with the added assurance of long-term support and maintenance directly from Neo4j.

Documentation

Documentation can be found [here](#)

First thought - chunks

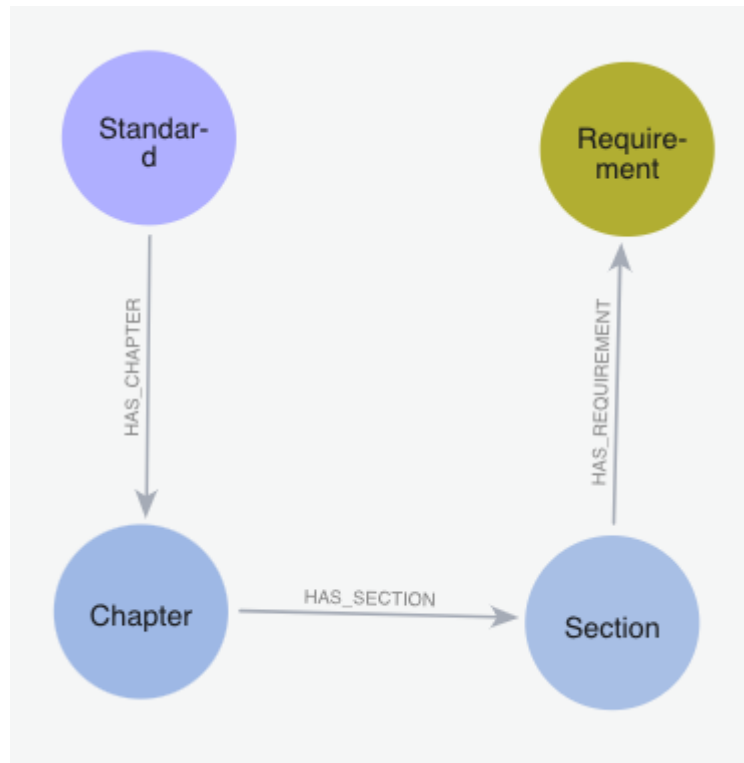


Taking advantage of the structure

- <https://medium.com/@irina.karkkanen/rag-on-graph-db-using-fixed-entity-architecture-make-you-retrieval-work-for-you-f4bfcac5277f>
- <https://medium.com/@irina.karkkanen/three-layer-fixed-entity-architecture-for-efficient-rag-on-graphs-787c70e3151a>
- Ontological fishbone!



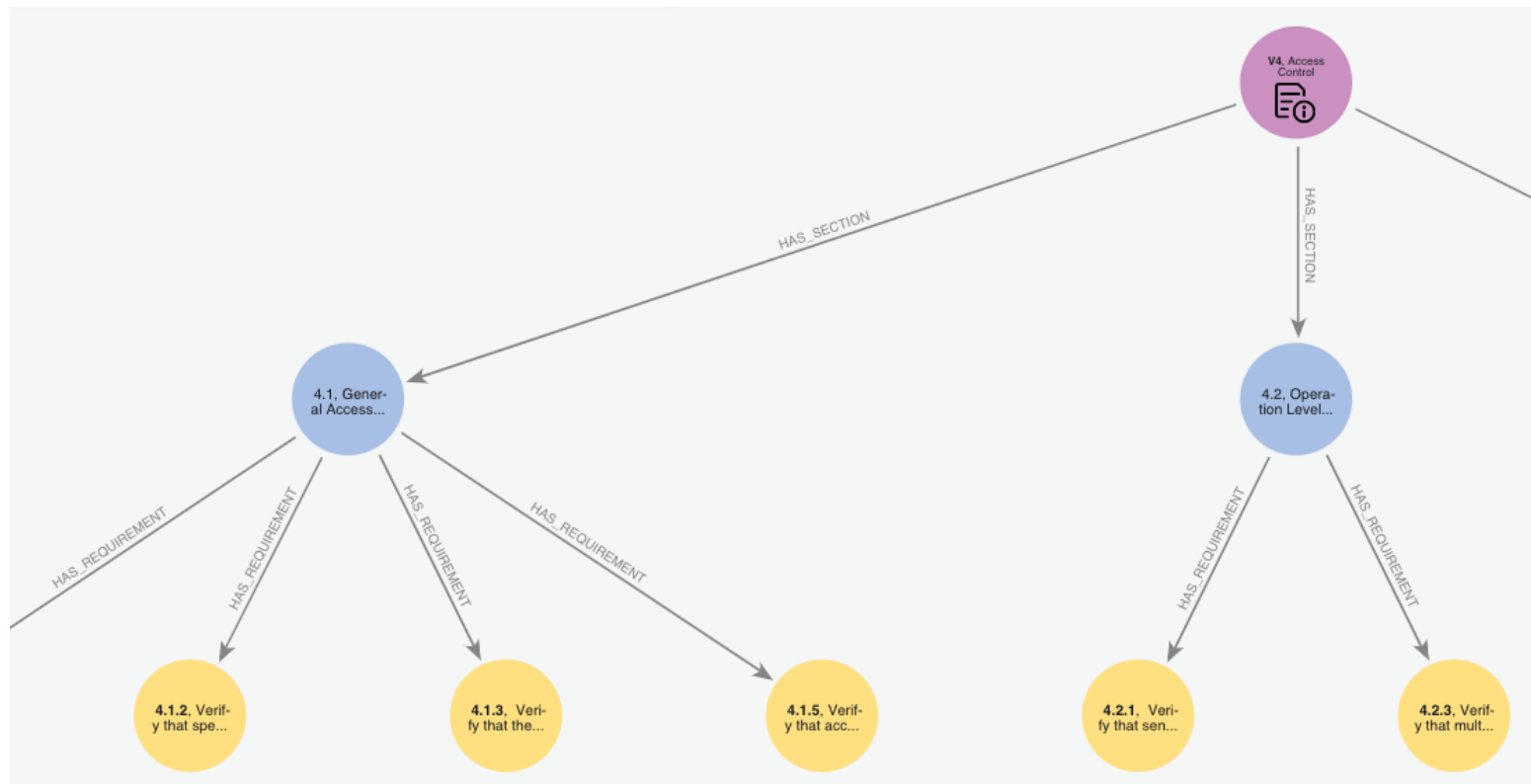
Ontological fishbone



In many organizations, subject matter experts can easily identify critical entities and relationships within well-defined domains. This foundational knowledge is essential for building effective knowledge graphs. By leveraging this expertise, a basic “fishbone” of key entities and their relationships can be established.



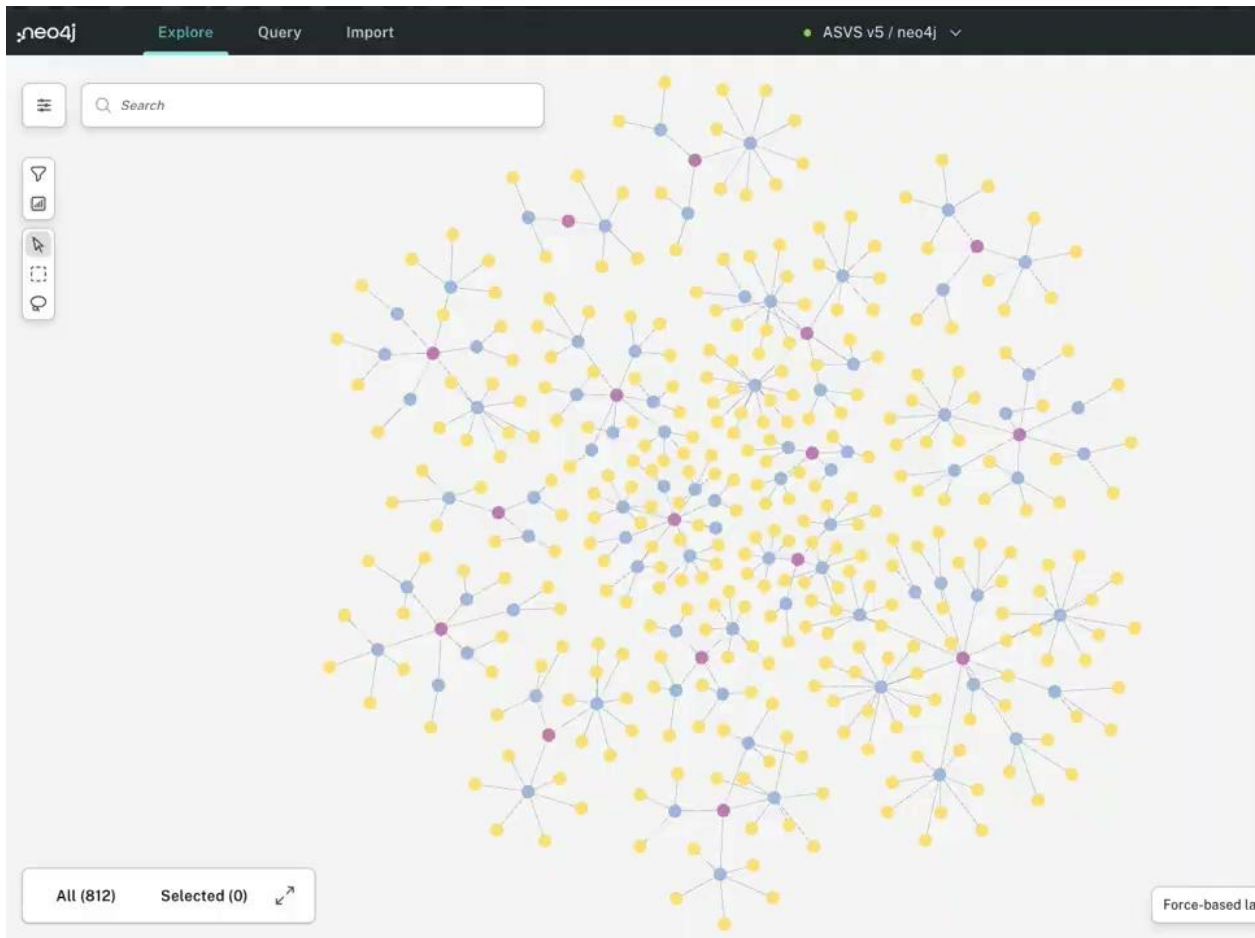
Next version



What's different?

- Uses ASVS V5
- Embedding is a property rather than a node
- Chapters, sections, requirements all have embeddings
- Uses EMBEDDING_MODEL = "text-embedding-3-small"
- Comes with full text indexes in addition to the vector index

<https://github.com/neo4j-examples/appsec-asvs-bot/tree/ASVS-v5>



```
MATCH (m1:Requirement{ID:'2.9.1'})  
  WITH m1, m1.embedding as e
```

```
// 2. Find other requirements which have high semantic similarity on description
```

```
CALL db.index.vector.queryNodes("embeddingIndex", 10, e) YIELD node, score
```

```
WITH m1, node AS m2, score
```

```
WHERE m2:Requirement and score < 1.0 and score > 0.80 // exclude self and low-scoring matches
```

```
// 3. For returned Requirement nodes, check they are in the same section and/or chapter
```

```
// Counting hops between the two requirements. 2 means same section, 4 means same chapter
```

```
WITH m1, m2, score
```

```
MATCH (m1)-[r*..6]-(m2)
```

```
// 4. Use structure to calculate weights and apply to similarity score
```

```
WITH m1, m2, score,
```

```
  CASE size(r)
```

```
    WHEN 2 THEN 2
```

```
    WHEN 4 THEN 1+log(2)
```

```
    ELSE 1
```

```
  END AS proximity
```

```
RETURN m2.ID AS reqNumber, m2.`Description` AS requirement, score,
```

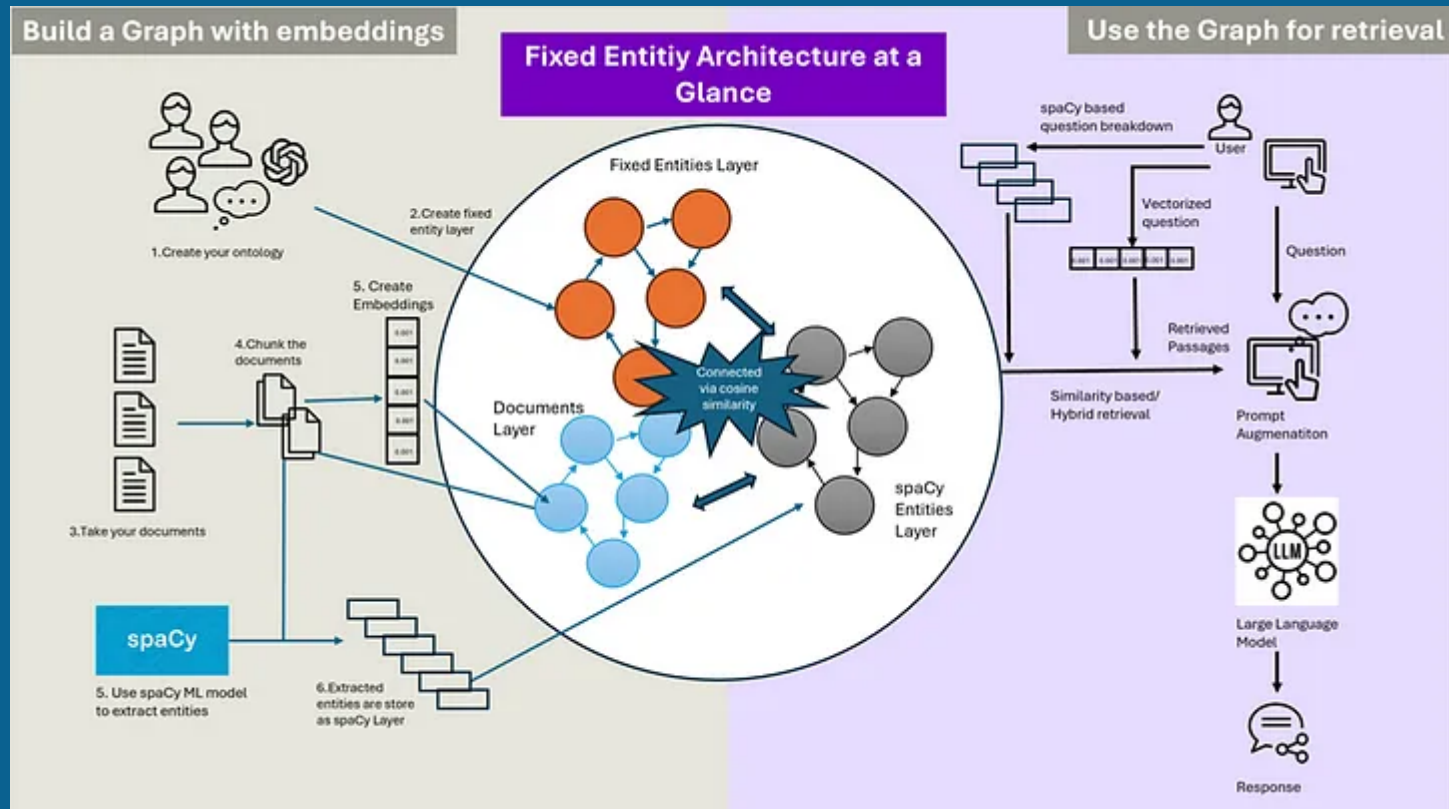
```
  proximity * score AS rank
```

```
ORDER BY rank DESC
```

Taking advantage of the structure

- <https://medium.com/@irina.karkkanen/rag-on-graph-db-using-fixed-entity-architecture-make-you-retrieval-work-for-you-f4bfcac5277f>
- <https://medium.com/@irina.karkkanen/three-layer-fixed-entity-architecture-for-efficient-rag-on-graphs-787c70e3151a>

What's next?



Thank you!

[https://github.com/neo4j-examples/
appsec-asvs-bot](https://github.com/neo4j-examples/appsec-asvs-bot)

irene.michlin@neo4j.com