

# Towards Hybrid Graphs: Unifying Property Graphs and Time Series

---

SHUBHANGI AGARWAL  
[sagarwal.mail@gmail.com](mailto:sagarwal.mail@gmail.com)



UNIVERSITÄT  
LEIPZIG



# Property Graphs (PG)

- Nodes and edges have labels and key-value properties
- Can store rich, contextual metadata (e.g., timestamps, weights, types)
- Supports complex queries using both structure and attributes

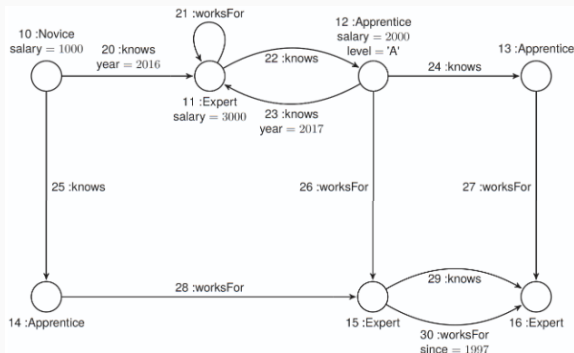


FIGURE 1: AN EXAMPLE OF A PROPERTY GRAPH

# Time Series

- Chronologically indexed data points
  - Key-value pairs, key is timestamp
- Captures state of an object over time
- Applications in various domains
  - Network monitoring, finance, healthcare, etc.

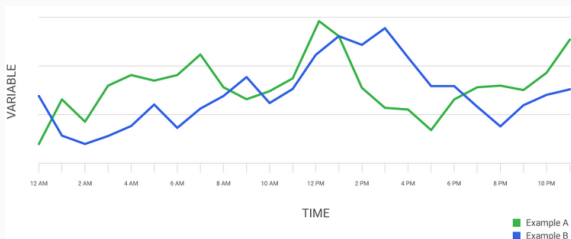


FIGURE 2: AN EXAMPLE OF A TIME SERIES

# The Missing Link

## Native Graph Databases

- Nodes directly linked/ adjacency lists
- Allows for low latency traversals (deep recursions)

## Time Series Databases

- Time based partitioning, Indexing - faster range queries
- Columnar storage - efficient aggregation, compression
- Time-aware features
  - Downsampled views and computational facilities – Append only writes, Retention policies

# The Missing link

## Why combine Graphs and Time Series?

- Richer context → Deeper analysis
- Unified Querying and Reasoning
- No context switch between systems

# The Missing link

## Why combine Graphs and Time Series?

- Richer context → Deeper analysis
- Unified Querying and Reasoning
- No context switch between systems

## Ongoing attempts at bridging the gap

- Time Series Clustering via Community Detection in Networks, *Ferreira and Zhao*, Information Sciences 2016
- Series2Graph: Graph-based Subsequence Anomaly Detection for Time Series, *Boniol and Palpanas*, VLDB 2020
- Managing data of sensor-equipped transportation networks using graph databases, *Bollen et al*, Geoscientific Instrumentation 2024

# HYGRAPH Vision <sup>[1]</sup>

## Data Model

- Seamless aggregation of data models
- Simplified and efficient storage, access, and processing

---

[1] Towards Hybrid Graphs: Unifying Property Graphs and Time Series, Ammar et al, EDBT 2025

# HYGRAPH Vision <sup>[1]</sup>

## Data Model

- Seamless aggregation of data models
- Simplified and efficient storage, access, and processing

## Analytical Pipeline

- Property Graph operations
- Time Series operations
- Hybrid operations - combined analysis
- Optimized algorithms

---

[1] Towards Hybrid Graphs: Unifying Property Graphs and Time Series, Ammar et al, EDBT 2025

# HYGRAPH Vision <sup>[1]</sup>

## Data Model

- Seamless aggregation of data models
- Simplified and efficient storage, access, and processing

## Analytical Pipeline

- Property Graph operations
- Time Series operations
- Hybrid operations - combined analysis
- Optimized algorithms

## Interface

- APIs, Library, Programming Language

---

[1] Towards Hybrid Graphs: Unifying Property Graphs and Time Series, Ammar et al, EDBT 2025

# PG Data Model <sup>[1]</sup>

Given a set of labels, property keys and values.

Property Graphs are formed by

- Vertex and edge set,  $V$  and  $E$
- Functions to map
  - edges to pair of vertices:  $\rho$
  - vertices and edges to labels:  $\lambda$
  - vertices and edges to set of property key-value pairs:  $\phi$

---

[1] The property graph database model, Renzo Angles, AMW 2018

# PG Data Model <sup>[1]</sup>

Given a set of labels, property keys and values.

Property Graphs are formed by

- Vertex and edge set,  $V$  and  $E$
- Functions to map
  - edges to pair of vertices:  $\rho$
  - vertices and edges to labels:  $\lambda$
  - vertices and edges to set of property key-value pairs:  $\phi$

Temporal Property Graphs (TPGs)

- vertices and edges to pair of timestamps (validity period):  $\nu$

---

[1] The property graph database model, Renzo Angles, AMW 2018

# PG Data Model <sup>[1]</sup>

Given a set of labels, property keys and values.

Property Graphs are formed by

- Vertex and edge set,  $V$  and  $E$
- Functions to map
  - edges to pair of vertices:  $\rho$
  - vertices and edges to labels:  $\lambda$
  - vertices and edges to set of property key-value pairs:  $\phi$

Temporal Property Graphs (TPGs)

- vertices and edges to pair of timestamps (validity period):  $\nu$

Concisely,

$$\mathcal{G} = \langle V, E, \rho, \lambda, \phi, \nu \rangle$$

---

[1] The property graph database model, Renzo Angles, AMW 2018

# Time Series Data Model

Mathematically,

$$ts = \{(t_1, y_1), (t_2, y_2), \dots, (t_n, y_n)\}$$

- $t_i$  is a timestamp
- $t_i < t_j$ , if  $i < j$
- $y_i = \langle val_1, val_2, val_3, \dots \rangle$  (multi-variate)

# HYGRAPH - Extending TPGs with Time Series

$$HG = \langle V, E, S, TS, \rho, \lambda, \phi, \nu, \psi, \delta \rangle$$

Additional notations,

- $S$ : set of dynamic subgraphs
- $TS$ : set of multi-variate time series
- $\psi$ : maps subgraph at given timestamp to constituent vertices and edges
- $\delta$ : time series vertices and edges to time series in  $TS$

# HYGRAPH - Extending TPGs with Time Series

$$HG = \langle V, E, S, TS, \rho, \lambda, \phi, \nu, \psi, \delta \rangle$$

Additional notations,

- $S$ : set of dynamic subgraphs
- $TS$ : set of multi-variate time series
- $\psi$ : maps subgraph at given timestamp to constituent vertices and edges
- $\delta$ : time series vertices and edges to time series in  $TS$

What's different?

- $V$  contains PG ( $V_{pg}$ ) and Time series ( $V_{ts}$ ) vertices
- Similarly,  $E = E_{pg} \cup E_{ts}$
- Property key-value pairs for  $V, E$  and  $S$
- Values of keys could be time series

# HYGRAPH - Twin Representation System

**PG nodes and edges with Time Series as properties**

**Time Series nodes and edges**

# HYGRAPH - Twin Representation System

## PG nodes and edges with Time Series as properties

- Querying over topology and time series attributes
  - Finance data augmented with various transaction details
  - Packets transmitted between network nodes

## Time Series nodes and edges

# HYGRAPH - Twin Representation System

## PG nodes and edges with Time Series as properties

- Querying over topology and time series attributes
  - Finance data augmented with various transaction details
  - Packets transmitted between network nodes

## Time Series nodes and edges

- Querying over time series which are semantically connected
  - Stock market analysis based on feature similarity
  - Weather analysis with geographical connections

# HYGRAPH- How it all works?

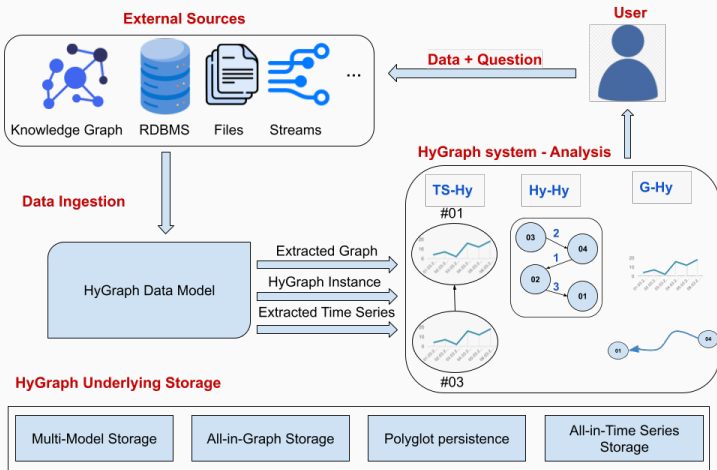
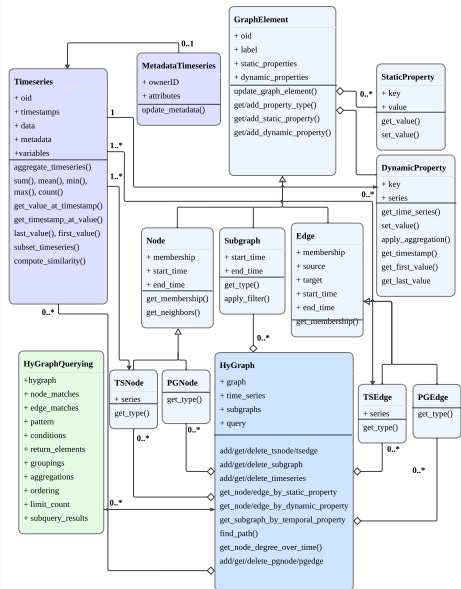


FIGURE 3: SCHEMATIC DIAGRAM OF VISIONED HYGRAPH DATA MODEL

# HYGRAPH- UML Design [1]

## HYGRAPH Module

- GraphElement Module
- Timeseries Module
- HyGraphQuerying Module
- Graph construction and batch loader module



[1] Transforming Time Series into Graphs and Back with HyGraph, Ammar et al, TGD Workshop, EDBT 2025

# HYGRAPH - Example Operations

## GraphSim

Graph construction from correlated time series

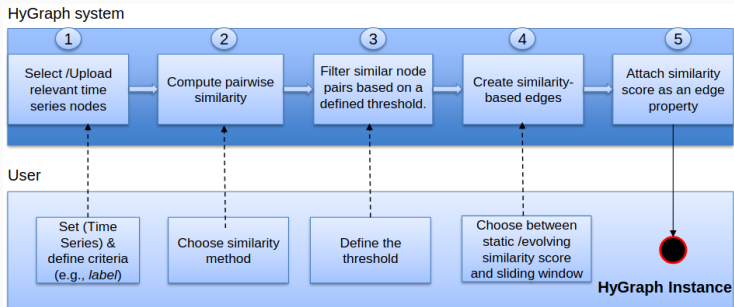


FIGURE 4: STEPS IN GRAPHSIM OPERATOR

# HYGRAPH - Example Operations

## ExtractTS

Extract time series from graph

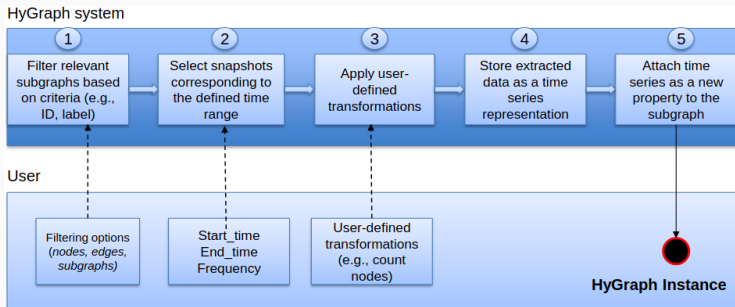


FIGURE 5: STEPS IN EXTRACTTS OPERATOR

# HYGRAPH - System Architecture

Open source Python library: **hygraph-core**

<https://pypi.org/project/hygraph-core/>

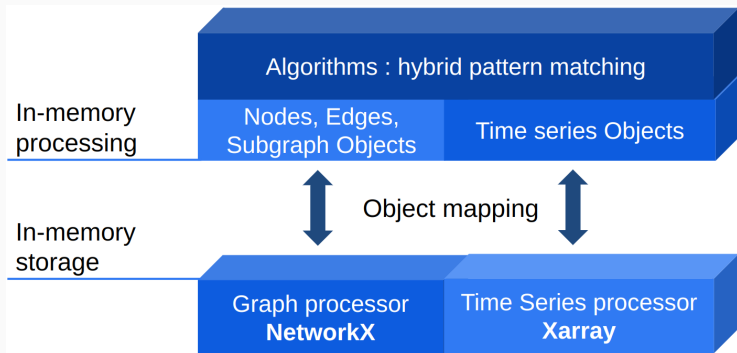


FIGURE 6: IMPLEMENTED ARCHITECTURE

# Road onwards...

- Different storage options, row vs columnar database
- Efficient querying of Time Series
  - Exploring indexing strategies
- Need for standardized query language for optimal use
- Development of hybrid operators

**Questions?**

**Thank you!**