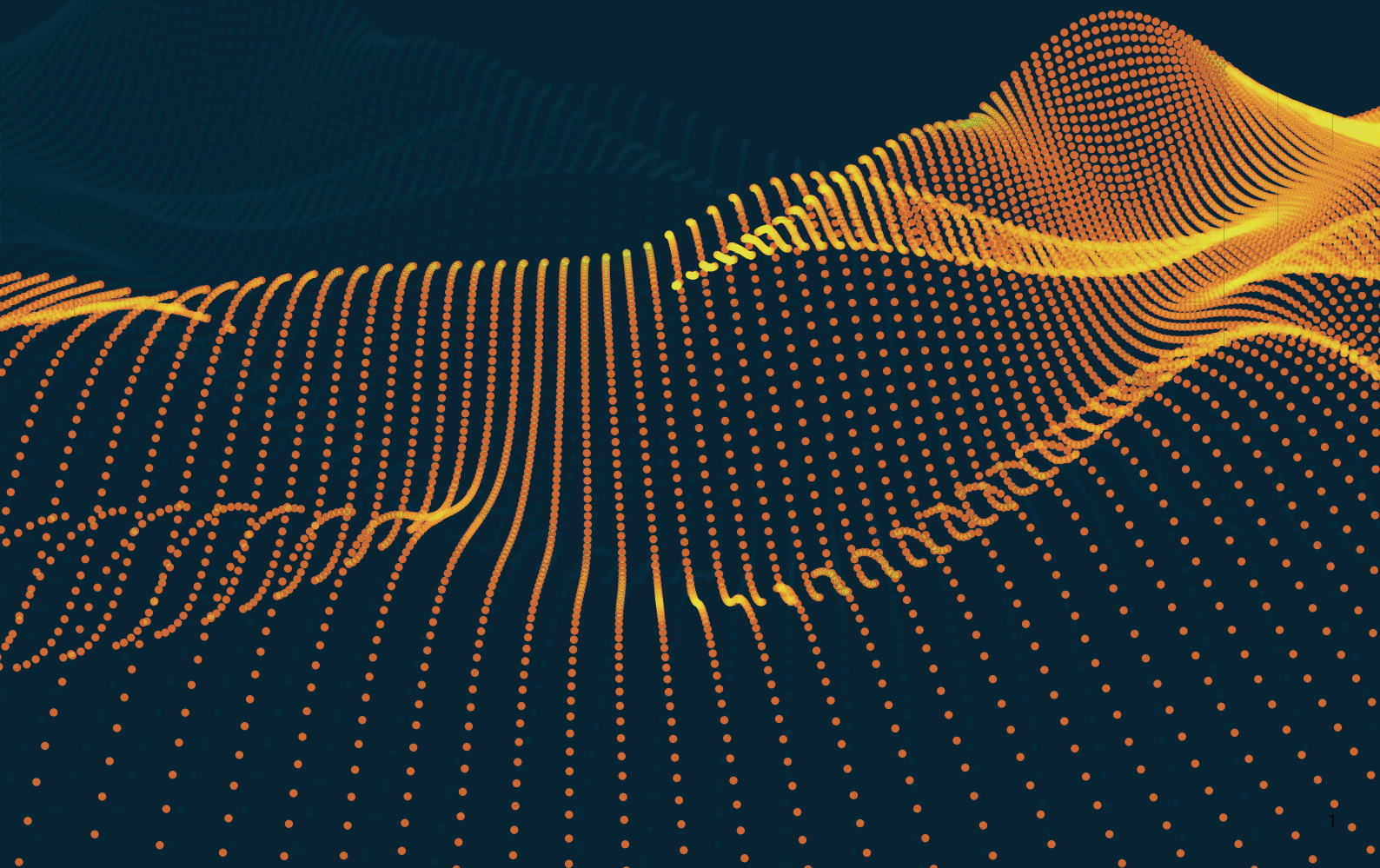
SmartDataLake

A Toolkit for Analytics over Data Lakes

github.com/smartdatalake

smartdatalake.eu

@SmartDataLake

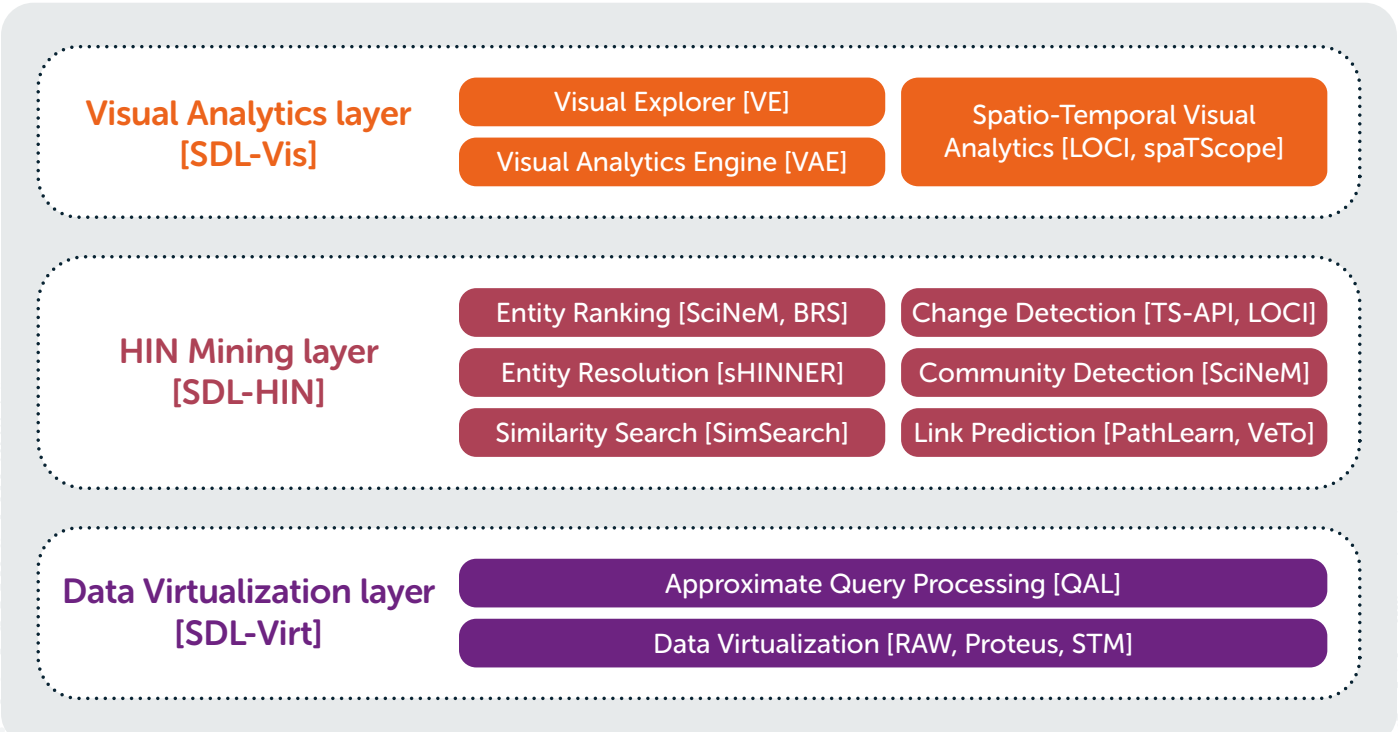


Overview

Data lakes are raw data ecosystems, where large amounts of diverse data are retained and coexist. They facilitate self-service analytics for flexible, fast, ad hoc decision making. SmartDataLake enables extreme-scale analytics over sustainable big data lakes.

- The toolkit provides an adaptive, scalable and elastic data lake management system that offers: (a) data virtualization for abstracting and optimizing access and queries over heterogeneous data, (b) data synopses for approximate query answering and analytics to enable interactive response times, and (c) automated placement of data in different storage tiers based on data characteristics and access patterns to reduce costs.
- The data lake's contents are modelled and organised as a heterogeneous information network, containing multiple types of entities and relations. Efficient and scalable algorithms are provided for: (a) similarity search and exploration for discovering relevant information, (b) entity resolution and ranking for identifying and selecting important and representative entities across sources, (c) link prediction and clustering for unveiling hidden associations and patterns among entities, and (d) change detection and incremental update of analysis results to enable faster analysis of new data.
- Interactive and scalable visual analytics are provided to include and empower the data scientist in the knowledge extraction loop. This includes functionalities for: (a) visually exploring and tuning the space of features, models and parameters, and (b) enabling large-scale visualizations of spatial, temporal and network data.

Components

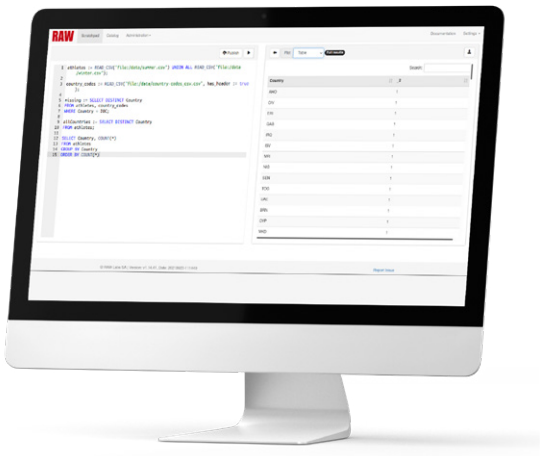


Data Virtualization Layer: SDL-Virt

SDL-Virt

The SDL-Virt layer is responsible for delivering efficient data access over a data lake containing large volumes of heterogeneous data. It encapsulates and abstracts all issues related to the efficient placement, distribution, management, and retrieval of data, providing to upper layers homogeneous access to data through an SQL-like query language. Specifically, the main offered functionalities include: (i) data virtualization over different data types and formats, (ii) automated placement of data over different storage tiers to optimize the tradeoff between storage cost and speed of retrieval, (iii) approximate query processing, that can speed up data analysis by enabling approximate answers with theoretical guarantees for accuracy.

RAW is a query engine that allows users to pose questions in a SQL-like language to their data without any previous processing. It copes with hierarchical data which are not usually supported in databases. It also supports multiple input locations, including HDFS, HTTP, Amazon S3, Dropbox and relational database systems, multiple input formats, including CSV, JSON, HJSON, XML, Microsoft Excel, log files, and multiple output formats, including JSON, HJSON, Parquet among others.



Proteus is a heterogeneous, just-in-time (JIT) compiled, in-memory data processing system that is used in SDL-Virt to accelerate analytical queries. Although JIT compilation has become popular for reducing interpretation overheads, existing systems parallelize execution based on the assumption of a CPU-only architecture. This offers the convenience of a cache-coherent shared memory and atomic operations on shared data structures. However, this execution model cannot work on heterogeneous systems where a system-wide cache coherence is not available. Proteus solves this problem, by redesigning the traditional *exchange* operator of the Volcano model and encapsulating parallelism across both CPUs and GPUs.

The Storage Manager (STM) is a component that enables the efficient vertical integration between RAW and Proteus and accelerates workloads with working-sets that exceed the capacity of main memory. An STM instance is installed in every node of the cluster and manages a locally available storage hierarchy. By using a set of pluggable policies, it automatically places each data chunk to the most appropriate storage medium. Despite being used only for RAW and Proteus in the context of SmartDataLake, STM exposes a generic object-store API, can support multiple concurrent clients, and can be employed as a multi-tiered external cache for any big data framework.

QAL is an approximate query engine that executes SQL aggregation queries over synopses which are compact summaries of the original data, allowing to provide approximate results bounded with user-defined confidence. QAL adapts construction of synopses to the workload such that it generates useful synopses ahead of time. Unlike the other state-of-the-art approximate query processing (AQP) engines that only utilize samples, QAL leverages sketches such as Count-min sketch with dyadic ranges, so that it can propose various execution plans for the approximate queries.

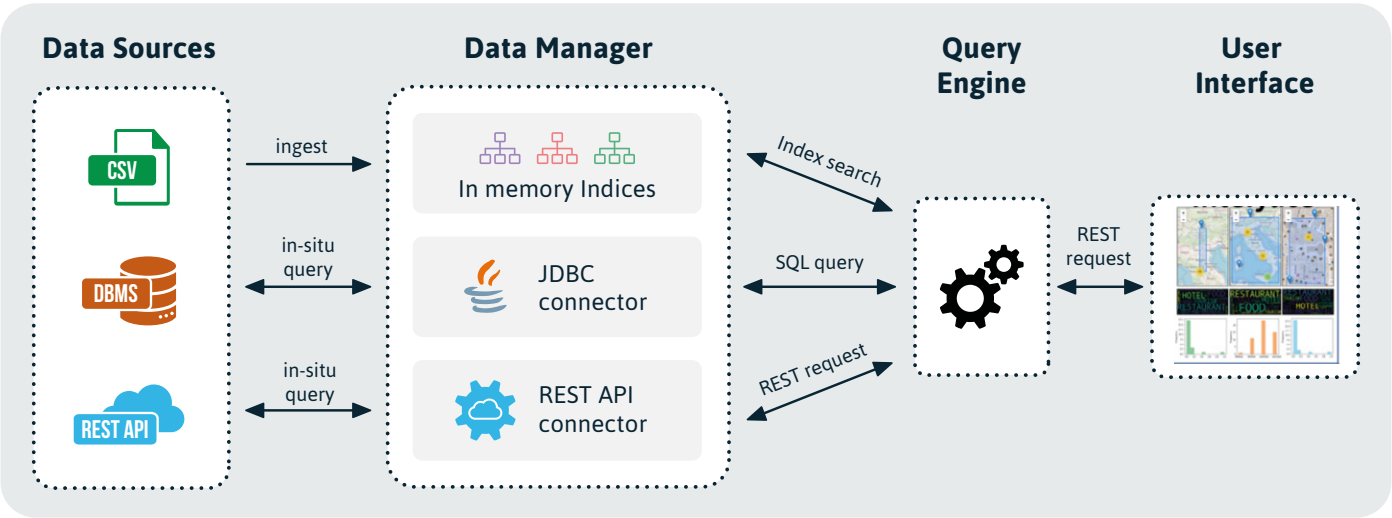
HIN Mining Layer: SDL-HIN

SDL-HIN

The SDL-HIN layer is dedicated to searching and analyzing the contents of a data lake, being represented in the form of a Heterogeneous Information Network (HIN), i.e., a graph consisting of entities and relations of different types. The offered functionalities include: (i) discovering similar or near-duplicate entities under different similarity criteria and matching conditions, (ii) ranking of entities based on the structure of the graph, as well as on other domain-specific criteria such as properties of geospatial regions, (iii) predicting or suggesting links between entities based on their attributes and position in the network, (iv) detecting communities of entities, including potentially overlapping or hierarchical communities, and (iv) detecting changes in evolving communities and in data represented as time series.

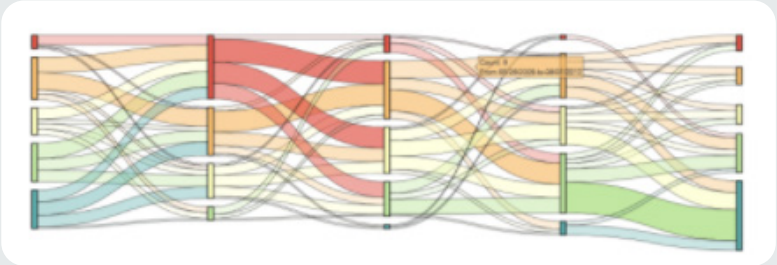
sHINNER is the component responsible for Entity Resolution (ER) in HINs. The functionalities of sHINNER are based on Graph Generating Dependencies (GGDs), a new class of graph dependencies proposed for property graphs. The GGDs allows the user to rewrite entity matching rules according to the entity topological information in the graph/HIN (graph pattern) and the similarity between its attributes (similarity constraints). Given the user input, sHINNER is responsible for identifying the entities that will be matched and "fix it" by generating new nodes/edges in the graph.

The **SimSearch** component supports top-k similarity search over multi-attribute entity profiles possibly residing in different, remote, or heterogeneous data sources. The similarity search queries may involve different similarity measures (Jaccard, Euclidean, Manhattan, etc.) against multi-attribute entities, i.e., datasets with different types of properties: categorical (set-valued), textual (string), numerical, spatial (i.e., locations), or temporal (date/time values). Attribute data values may come from diverse data sources, and each one can be either ingested from CSV files or queried in-situ from a DBMS (like Proteus or PostgreSQL) or available from REST APIs (like JSON data hosted in Elasticsearch). For ingested data, suitable indices are constructed in memory (e.g., R-trees for spatial locations, B-trees for numerical values, inverted indices for sets of textual values).



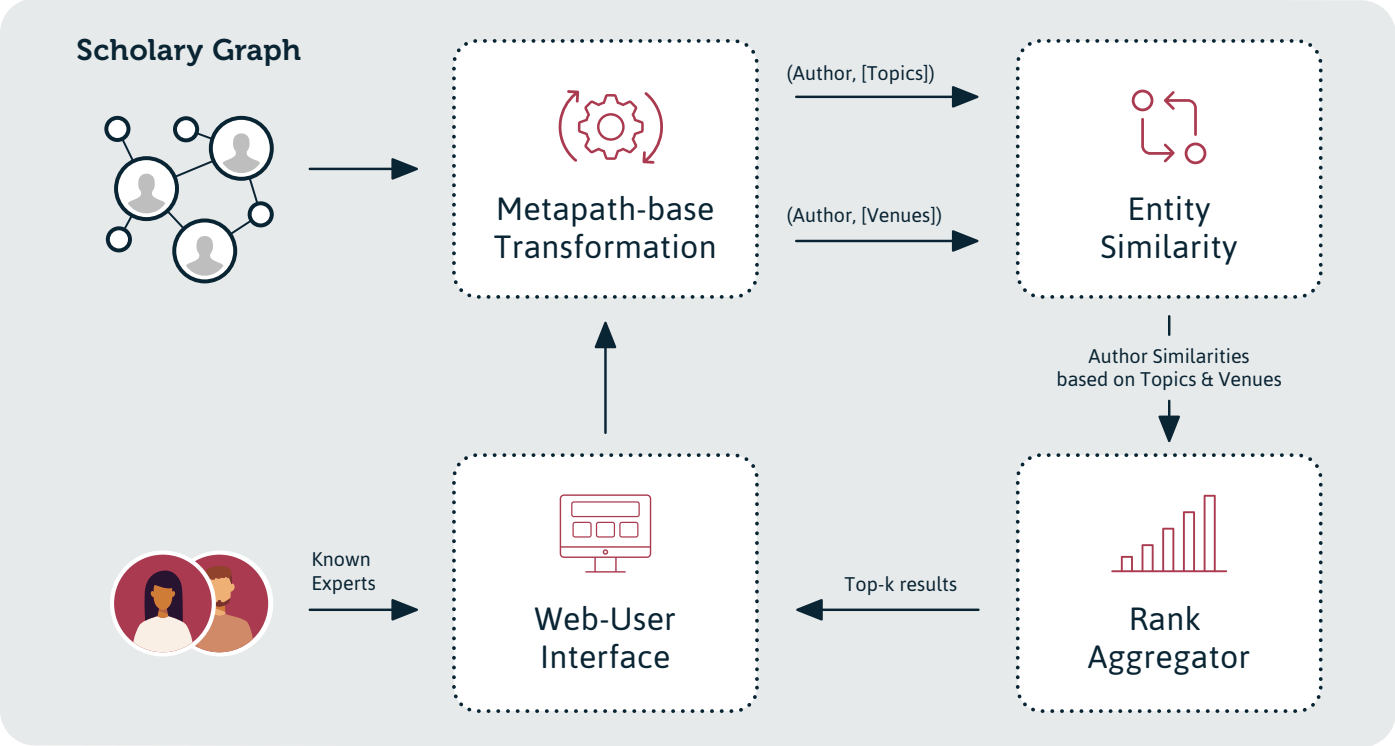
The **TS-API** supports multiple and single time series analytics, including forecasting, correlation analysis, bundle discovery, self-join, change point detection and seasonality decomposition The component can assist data scientists to analyze and obtain useful insights from either a single time series, or from time series sets.

LOCI provides a suite of functionalities for analysing, mining, and visualizing spatial and temporal data. More specifically, it offers functionalities for spatial exploration and mining over Points and Areas of Interest, as well as for change detection and seasonality decomposition in time series data, and evolution tracking of dynamic sets of entities.



PathLearn performs link prediction in hererogeneous information networks by modeling the effect of every path that exists between pairs of nodes, and aggregating all effects to estimate the probability that a link exists between them, taking into account their node/edge types and features. The implementation supports four functions that are required for using the model: (i) preprocessing, (ii) training, (iii) testing and (iv) prediction.

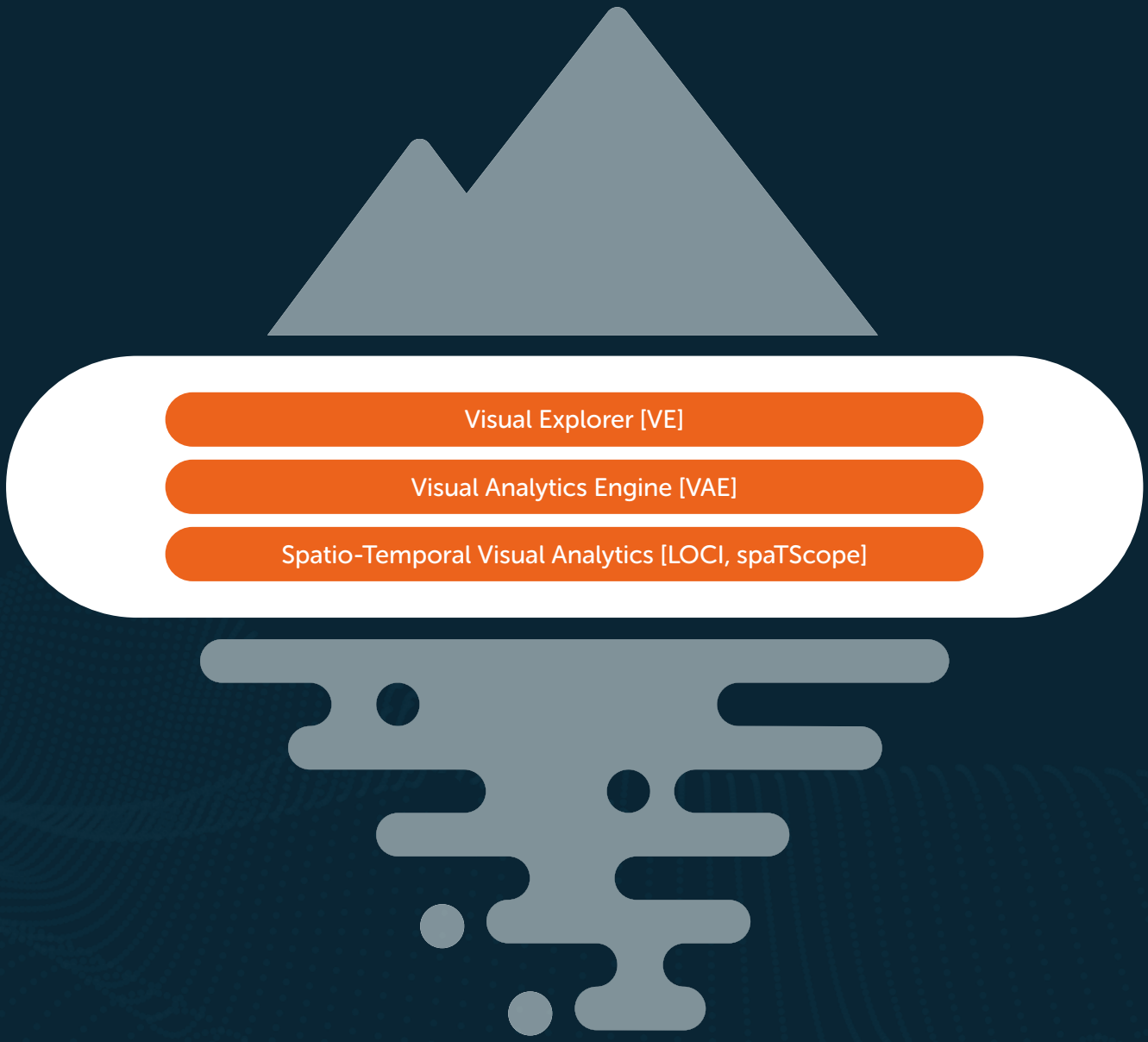
The **VeTo** component expands a set of given entities with new ones of the same type considering their structural similarity in a Heterogeneous Information Network (HIN). VeTo offers a Web user interface that provides expansion recommendations for a given expert set. A prototype version is deployed on top of AMiner's DBLP Citation Network Dataset and is publicly available to demonstrate VeTo's effectiveness in expert set expansion applications for academics.



The **BRS** component provides the top-k best region search on 2-dimensional data (e.g., maps). Specifically, BRS allows users to rank fixed-size regions that maximize a user-defined scoring function. By ranking hot regions and plotting results on maps, BRS helps data analyzers to explore data and extract knowledge. Identifying the best regions in a large-scale dataset becomes expensive, thus BRS proposed three approaches to distribute the problem of best region search over a Spark cluster.

SciNeM is an open-source tool that offers a wide range of functionalities for exploring and analysing HINs and utilises Apache Spark for scaling out through parallel and distributed computation. SciNeM provides an intuitive, Web-based user interface to build and execute complex constrained metapath-based queries and to explore and visualise the corresponding results. Under the hood, all the supported state-of-the-art HIN analysis types have been implemented in a scalable manner supporting the distributed execution of analysis tasks on computational clusters. SciNeM has a modular architecture making it easy to extend it with additional algorithms and functionalities. Currently, it supports the following operations, given a user-specified metapath: ranking entities using a random walk mode, retrieving the top-k most similar pairs of entities, finding the most similar entities to a query entity, and discovering entity communities.

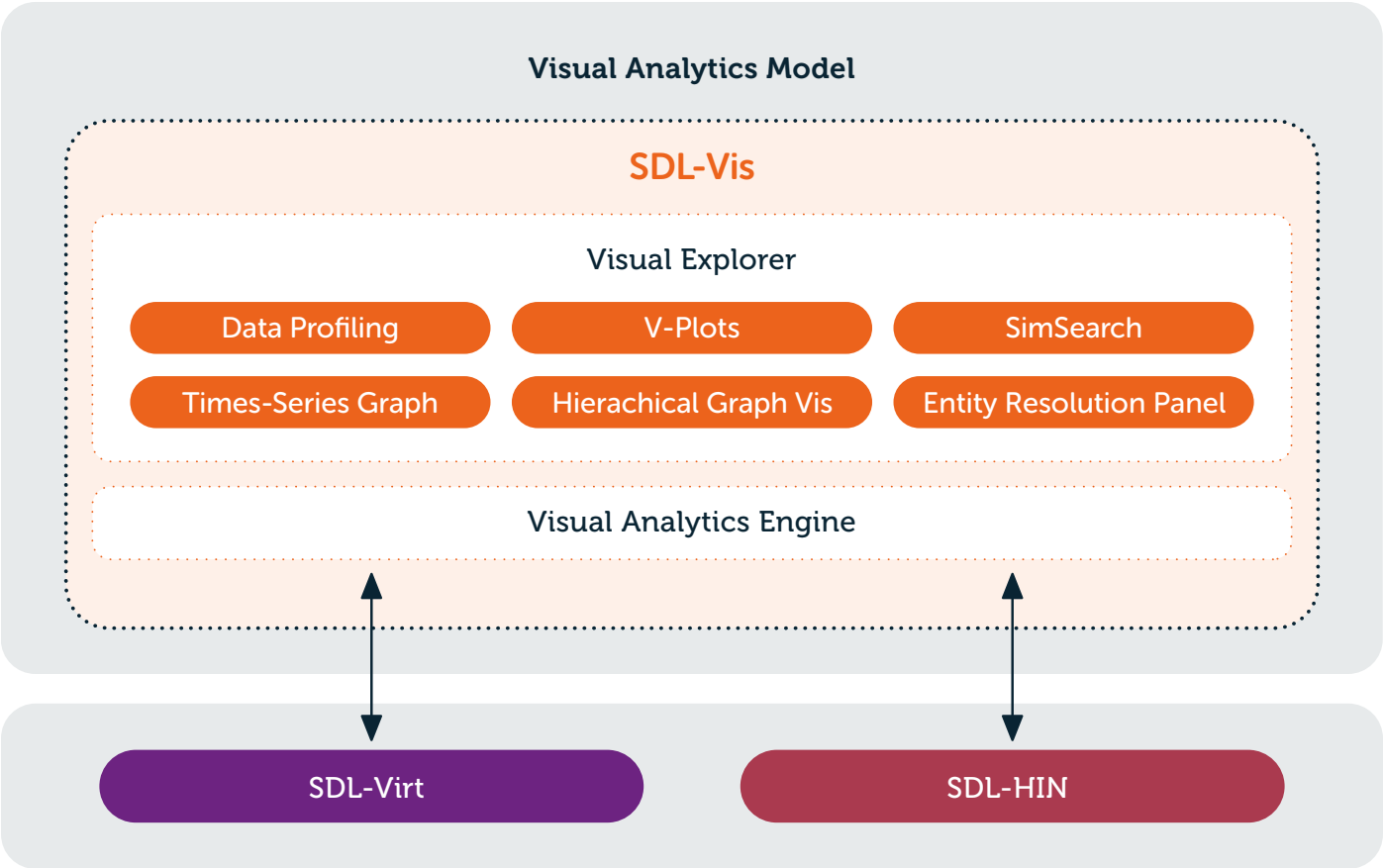
Visual Analytics Layer: [SDL-Vis]



SDL-Vis

The SDL-Vis layer includes the human in the data analysis loop through offering visual analytics capabilities building on top of the functionalities implemented in the two lower layers. Using a visual analytics model at its core to determine user interactions, it offers visualizations for supporting data profiling and parameter tuning. It also offers custom visualizations for specific types of data, including graph, spatial, and temporal.

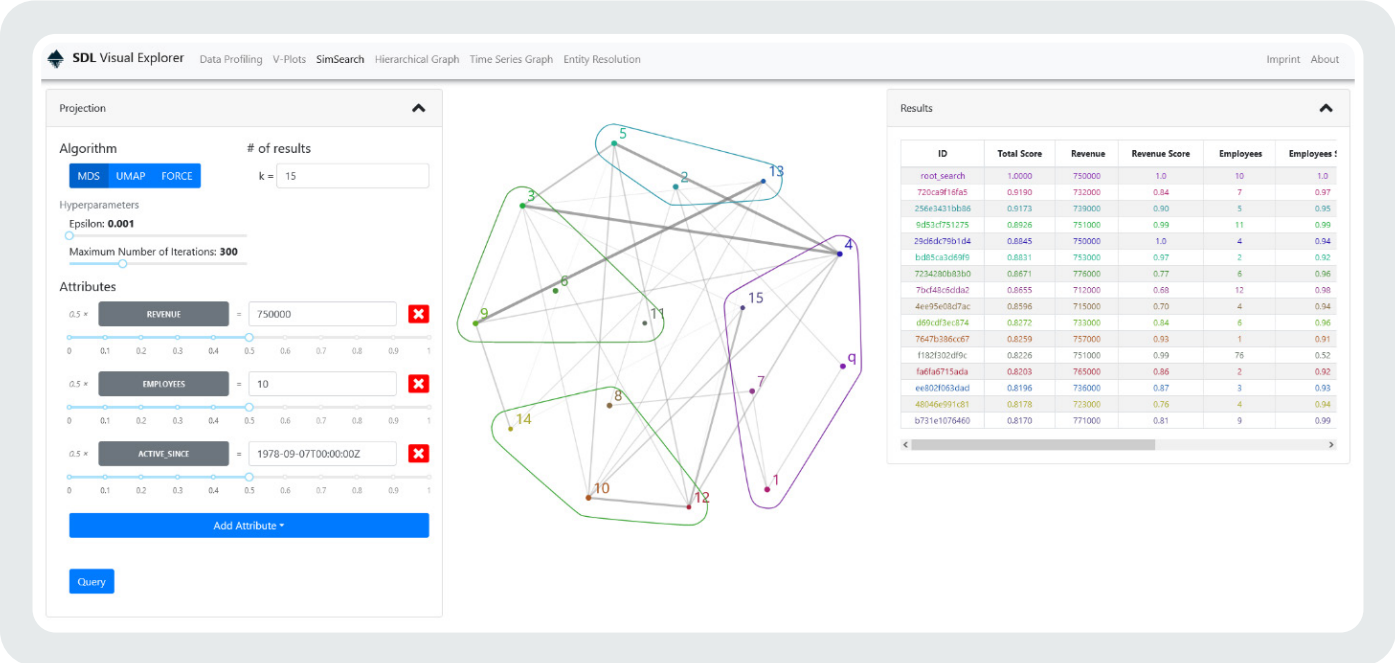
The Visual Explorer provides a graphical user interface that bundles multiple visual analytics applications in the context of SmartDataLake, enabling knowledge generation on the data and algorithms of SmartDataLake. Thereby, the Visual Explorer supports various relevant tasks when exploring large, heterogeneous data, such as data profiling, descriptive analytics, similarity search, hierarchical graph exploration, and entity resolution.



Data Profiling – The Data Profiling application provides access to a Jupyter Lab environment, which is pre-configured for scientific data analysis and visualization. It bundles examples on customized data profiling workflows and demonstrates how the user can connect to the components of SDL-Virt using different types of database adaptors. Furthermore, it shows how the API endpoints of the Visual Analytics Engine can be utilized for customized data profiling and visualization tasks.

The SmartDataLake visual analytics layer provides data profiling with the **V-Plots** application. A combined chart called a "V-Plot," provides a task-driven solution for descriptive analysis over numerical data columns. An interactive guide allows the user to enter her preferences and goals for the analysis. The V-Plots system translates those into a selection of suitable chart types to reach the analysis goals. The most suitable chart types are then automatically combined into a V-Plot.

SimSearch Projection – The SimSearch Projection enables interactive parameter tuning on the similarity search algorithm of SDL-HIN. It displays the search results and indicates how the specified parameters influence them. Furthermore, we facilitate real-time exploration of the local parameter space by pre-computing results for possible future user actions.

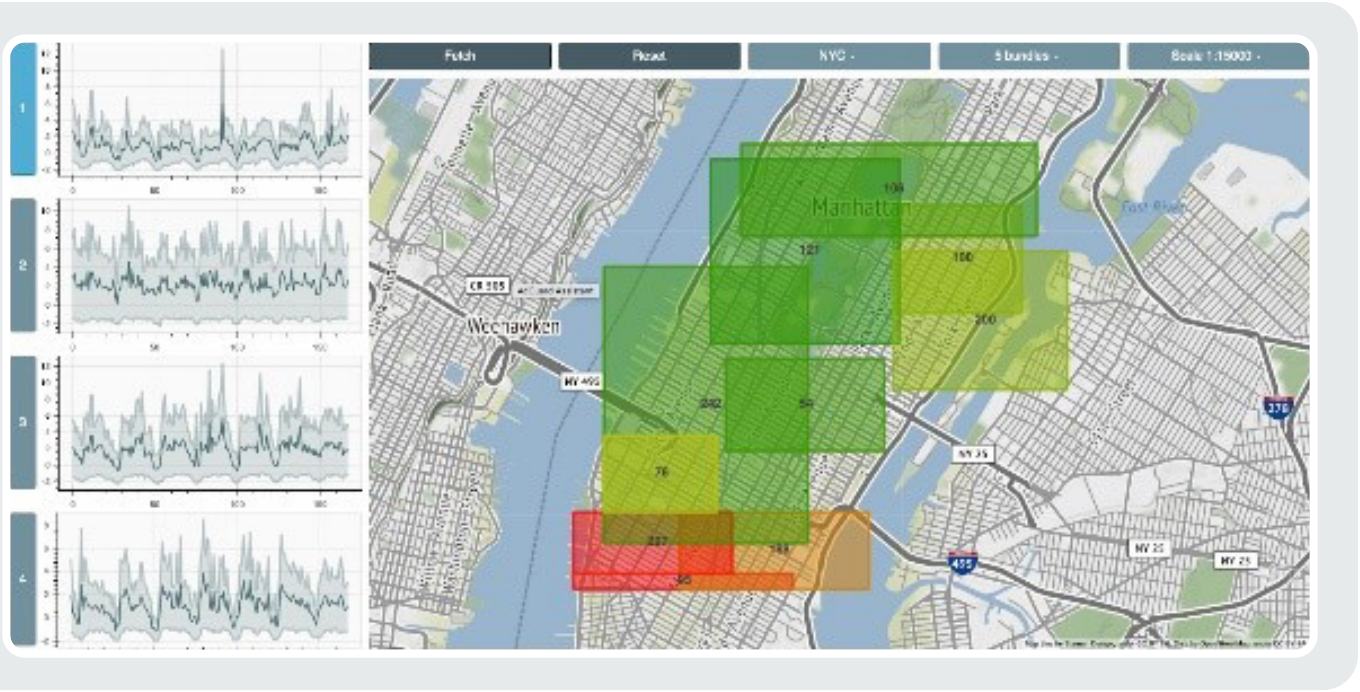


Entity Resolution – The entity resolution panel provides access to the entity resolution functionalities of sHINNER in SDL-HIN. It assists the user on how to set the GGDs and shows a sample of the matched entities. This panel also displays information about the graphs available in SDL, such as schema and properties of nodes and edges. The user can also visualize a sample of query results by submitting a graph query written in G-Core language.

Hierarchical Graph – The Hierarchical Graph Visualization builds upon the hierarchical cluster functionality of SDL-HIN's sHINNER component. It allows to query the clustering algorithm with different parameters and interactively explore the results. By expanding clusters, inner data points of interest can be tracked without losing the global context. This paradigm prevents information overload by implementing „overview first, details on demand," allowing interactive exploration of large graphs.

Time-Series Graph – With the Time-Series Graph, we enable both local and global exploration of correlations between time-series. It provides access to the time series correlation algorithm, which is part of SDL-HIN. On a global level, we display aggregated correlations between multiple time-series as links in a force-directed graph. Complementary, a bar chart indicates point-wise correlations between a selected pair of time-series for local investigation.

spaTScope assists data scientists to obtain useful insights regarding the spatio-temporal behaviour of multiple geolocated time series, such as in a water consumption scenario, the user can intuitively gain insights regarding the consumption behavior in specific neighborhoods. The component achieves this though exploring interactive geolocated time series through the application of visual exploration application. It also works on large geolocated, co-evolving, time series datasets.



Summary

The SmartDataLake toolkit offers a full stack of components for exploring and analyzing data in a data lake, aiming to facilitate data analysts and data scientists through their journey from raw data to actionable insights. In particular, SDL-Virt provides tools for efficient, scalable and streamlined access to large volumes of raw data based on data virtualization, automated management of multiple storage tiers, and approximate query processing over data summaries. SDL-HIN allows to explore and analyze entities in the data lake represented in the form of a heterogeneous information network, including functionalities for similarity search, entity resolution, entity ranking, link prediction, community detection, and change detection. SDL-Vis supports the human in the loop, by offering visual analytics over different types of data, including tabular, graph, spatial and temporal data.

Click on each component for their respective repository

**Proteus/STM are not public yet*

Visual Analytics layer [SDL-Vis]

- Visual Explorer [VE]
- Visual Analytics Engine [VAE]
- Spatio-Temporal Visual Analytics [LOCI, spaTScope]

HIN Mining layer [SDL-HIN]

- Entity Ranking [SciNeM, BRS]
- Entity Resolution [sHINNER]
- Similarity Search [SimSearch]
- Change Detection [TS-API, LOCI]
- Community Detection [SciNeM]
- Link Prediction [PathLearn, VeTo]

Data Virtualization layer [SDL-Virt]

- Approximate Query Processing [QAL]
- Data Virtualization [RAW, Proteus, STM]

Project Website *Publications* *GitHub*

SmartDataLake

This project has received funding from the European Union's
Horizon 2020 research and innovation programme under
grant agreement No 825041