

List of Project Topics

Last update: Apr 26, 2026

LDE Projects in the Context of Our Three Open-source Projects



▪ DAPHNE EU-project

<https://github.com/daphne-eu/daphne>

- Focus on integrated data analysis pipelines
- Project implementation mainly in C++



▪ Apache SystemDS

<https://github.com/apache/systemds>

- Focus on the end-to-end data science lifecycle
- Project implementation mainly in Java and DML



▪ TerseTS

<https://github.com/cmcuza/TerseTS>

- Focus on time series compression
- Project implementation mainly in Zig

Topics in DAPHNE

Implementation mainly in C++ and DaphneDSL, some also in Python

<https://github.com/daphne-project/daphne/issues?q=is%3Aissue%20state%3Aopen%20label%3A%22LDE%20summer%202026%22>

#1002 Automatic Fusion of Result Data Analysis Into Linear/Relational Algebra Operators



■ Motivation

- Data characteristics (e.g., sparsity, data distribution, sort orders, symmetry, ...) can be used to optimize a data analysis pipeline, e.g., through simplification rewrites, data repr selection, and algorithm selection
- Knowing the data characteristics of intermediate results in complex programs is difficult, since it either relies on (often inaccurate) compile-time estimates or on (expensive) run-time analyses

■ Task (in C++)

- Reduce the cost of run-time data analysis by fusing the analysis logic into the operator that produces the data
- As the set of relevant data characteristics depends on the context, a particular operator could be combined with any subset of data characteristics, making a manual implementation infeasible
- How can we automatically enhance given linear/relational algebra operators by data analyses of the output through code generation at the level of source code and/or the level of the compiler's intermediate repr?

■ More Information & Hints

- <https://github.com/daphne-eu/daphne/issues/1002>
- Contact: Patrick Damme

#939 Efficient Join Ordering Using the UES Method



■ Motivation

- Finding an efficient join order is one of the most important and most challenging tasks in relational query optimization and various methods have been proposed in the literature
- The UES method, which is based on upper bounds and sampling, is particularly simple yet effective

■ Task (in C++)

- Implement the UES method for join enumeration as a compiler pass in DAPHNE
- Evaluate the pass on well-known community benchmarks (e.g., the Join Order Benchmark)

■ More Information & Hints

- <https://github.com/daphne-eu/daphne/issues/939>
- Contact: Patrick Damme

#511 Efficient Parallel Hash-Join Operator for Speeding up the SSB



■ Motivation

- The Star Schema Benchmark (SSB) is a well-known benchmark for analytical query processing.
- The runtime of most SSB queries is dominated by PK-FK joins and semi-joins.
- An efficient join implementation is crucial for achieving good results in this benchmark.

■ Task (in C++)

- Implement an efficient parallel hash-join (separate build and probe) operator on columnar data.
- Devise an efficient hash table implementation and support it for intermediate results in DaphneIR.
- Parallelism should be achieved through multiple threads (MIMD), and optionally also through SIMD operations.
- In a first step, multi-threading could be applied inside the operator; based upon that, an integration with DAPHNE's vectorized engine could be tackled.

■ More Information & Hints

- <https://github.com/daphne-eu/daphne/issues/511>
- Contact: Patrick Damme

#986 DAPHNE and the Python Data Science Ecosystem: Efficient String Data Transfer



■ Motivation

- As Python is the language of choice for most data scientists nowadays, DAPHNE supports efficient bi-directional data transfer with popular Python libraries like numpy, pandas, SciPy, TensorFlow, and PyTorch
- So far, only the transfer of numeric data is supported
- However, real-world data sets often contain string-valued attributes that need to be transformed to numbers before applying ML algorithms and this transformation could happen either in Python or in DAPHNE

■ Task (in C++ and Python)

- Extend DAPHNE's data transfer to/from Python (e.g., numpy, pandas) by efficient support for string data
- Investigate if performing typical feature transformations like recoding and one-hot-encoding is more efficiently done in Python or in DAPHNE and optimize the transformations in DAPHNE

■ More Information & Hints

- <https://github.com/daphne-eu/daphne/issues/986>
- Contact: Patrick Damme

#985 Efficient File I/O Plug-ins for Widely-used File Formats



■ Motivation

- The input to integrated data analysis pipelines, that combine query processing, machine learning, and high-performance computing, could be provided in various general-purpose and domain-specific file formats
- To embrace a large variety of such formats, DAPHNE is extensible w.r.t. to file readers/writers, i.e., expert users can add their own file I/O plug-ins without touching the source code of DAPHNE
- However, so far there is only a limited number of plug-ins available

■ Task (in C++)

- Implement file I/O plug-ins for a range of widely used file formats for different data modalities, such as tabular data, (sparse) matrices/graphs, audio, images, and time series
- These plug-ins may be based on existing open-source libraries (with compatible license)
- Apply format-specific and format-agnostic tricks to make the file I/O efficient, e.g., by exploiting parallelism or pushing down certain operations into the readers

■ More Information & Hints

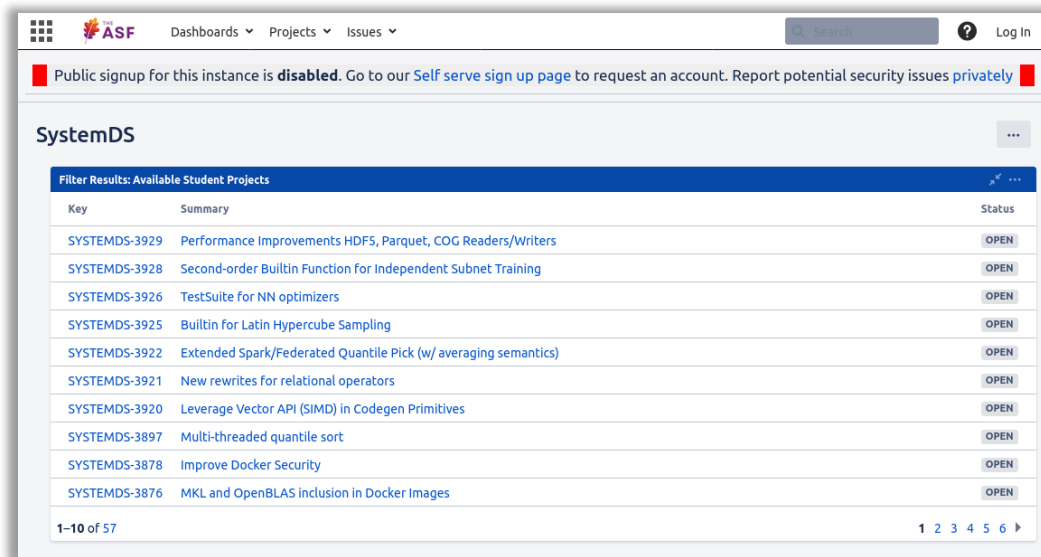
- <https://github.com/daphne-eu/daphne/issues/985>
- Contact: Patrick Damme

Topics in Apache SystemDS

Implementation mainly in Java and DML

<https://issues.apache.org/jira/secure/Dashboard.jspa?selectPageId=12335852#Filter-Results/12365413>

- See the **Full List** of Available Student Projects:
 - <https://issues.apache.org/jira/secure/Dashboard.jspa?selectPageId=12335852#Filter-Results/12365413>



The screenshot shows the Apache JIRA dashboard for the SystemDS project. The page title is "SystemDS" and the filter is "Filter Results: Available Student Projects". The table lists 10 issues with their keys, summaries, and "OPEN" status buttons.

Key	Summary	Status
SYSTEMDS-3929	Performance Improvements HDF5, Parquet, COG Readers/Writers	OPEN
SYSTEMDS-3928	Second-order Builtin Function for Independent Subnet Training	OPEN
SYSTEMDS-3926	TestSuite for NN optimizers	OPEN
SYSTEMDS-3925	Builtin for Latin Hypercube Sampling	OPEN
SYSTEMDS-3922	Extended Spark/Federated Quantile Pick (w/ averaging semantics)	OPEN
SYSTEMDS-3921	New rewrites for relational operators	OPEN
SYSTEMDS-3920	Leverage Vector API (SIMD) in Codegen Primitives	OPEN
SYSTEMDS-3897	Multi-threaded quantile sort	OPEN
SYSTEMDS-3878	Improve Docker Security	OPEN
SYSTEMDS-3876	MKL and OpenBLAS inclusion in Docker Images	OPEN

- Topics **Newly Added** in SoSe 2026
 - #3943 Compressed Federated Broadcast
 - #3942 LLM-assisted Generation of Data Augmentation Pipelines
- **Plus Many More Topics, e.g.**
 - #3929 Performance Improvements HDF5, Parquet, COG Readers/Writers
 - #3926 TestSuite for NN optimizers
 - #3925 Builtin for Latin Hypercube Sampling
 - #3922 Extended Spark/Federated Quantile Pick (w/ averaging semantics)
- **More Information & Hints**
 - Contact: Matthias Boehm

Topics in TerseTS

Implementation mainly in Zig

<https://github.com/cmcuza/TerseTS/issues>

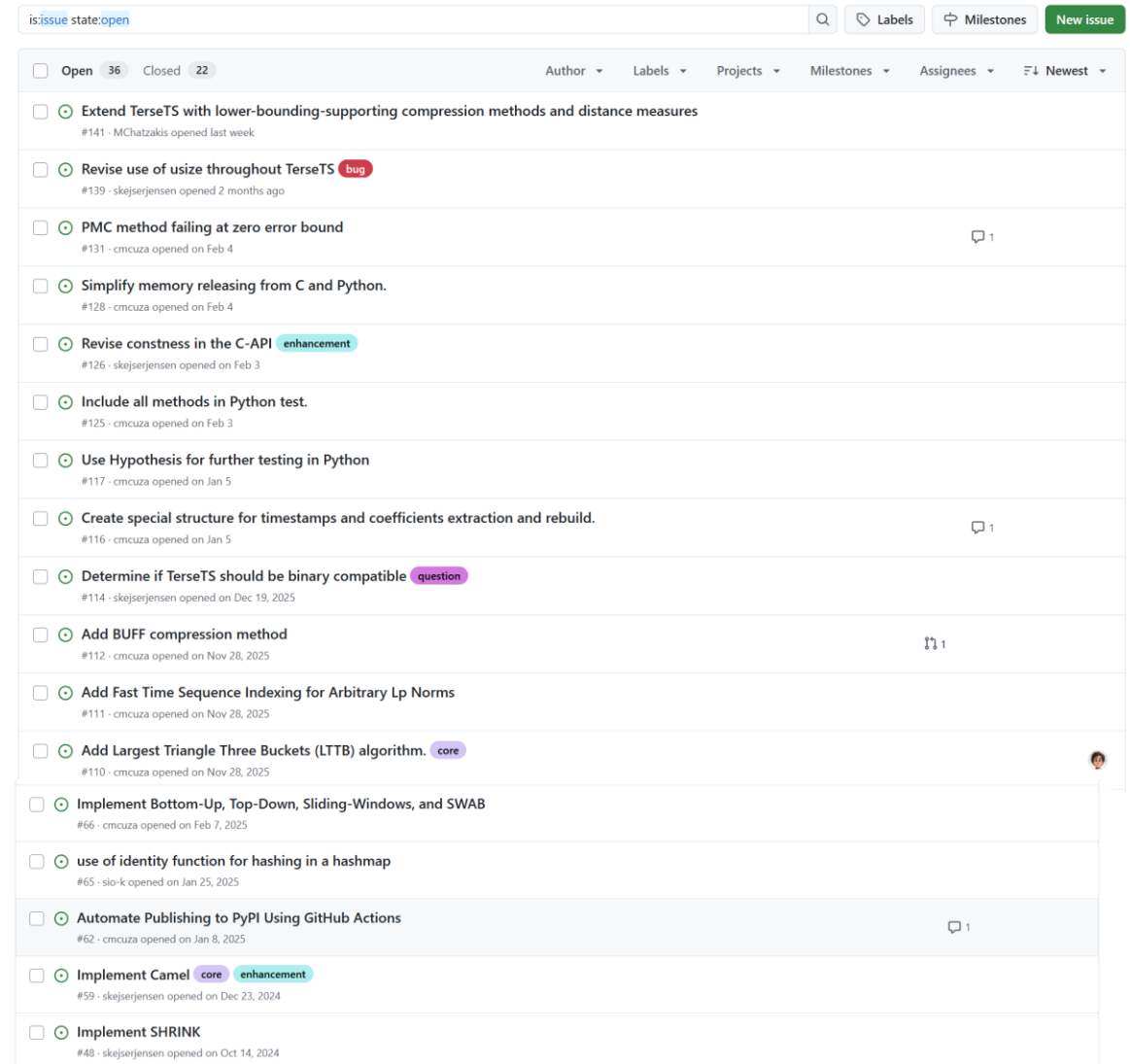
Topics in TerseTS

■ New Compression Algorithms or Tests/Benchmarks

- #141 Extend TerseTS with lower-bounding-supporting symbolic approximation methods
- #112 Extend TerseTS with BUFF algorithm using SIMD-instructions
- #29 Extend TerseTS with ALP algorithm using SIMD-instructions
- Extend TerseTS with at least three of the following algorithms: Largest Triangle Three Buckets (LTTB) (#110), GORILLA (#21), CHIMP (#22), CAMEL (#59), ELF (#28), SerfXOR (#142)
- #65,75,92,102,125 Improve TerseTS test-suite and benchmarking

■ More Information

- <https://github.com/cmcuza/TerseTS/issues>
- Contact: Carlos Enrique Muñiz-Cuza



The screenshot shows a GitHub repository page for 'TerseTS' with the filter 'is:issue state:open'. The issues are listed in a table with columns for 'Open' (36), 'Closed' (22), 'Author', 'Labels', 'Projects', 'Milestones', 'Assignees', and 'Newest'. The issues are sorted by newest. The visible issues are:

- #141 - MChatzakos opened last week: Extend TerseTS with lower-bounding-supporting compression methods and distance measures
- #139 - skejserjensen opened 2 months ago: Revise use of usize throughout TerseTS (bug)
- #131 - cmcuza opened on Feb 4: PMC method failing at zero error bound
- #128 - cmcuza opened on Feb 4: Simplify memory releasing from C and Python.
- #126 - skejserjensen opened on Feb 3: Revise constness in the C-API (enhancement)
- #125 - cmcuza opened on Feb 3: Include all methods in Python test.
- #117 - cmcuza opened on Jan 5: Use Hypothesis for further testing in Python
- #116 - cmcuza opened on Jan 5: Create special structure for timestamps and coefficients extraction and rebuild.
- #114 - skejserjensen opened on Dec 19, 2025: Determine if TerseTS should be binary compatible (question)
- #112 - cmcuza opened on Nov 28, 2025: Add BUFF compression method
- #111 - cmcuza opened on Nov 28, 2025: Add Fast Time Sequence Indexing for Arbitrary Lp Norms
- #110 - cmcuza opened on Nov 28, 2025: Add Largest Triangle Three Buckets (LTTB) algorithm. (core)
- #66 - cmcuza opened on Feb 7, 2025: Implement Bottom-Up, Top-Down, Sliding-Windows, and SWAB
- #65 - sio-k opened on Jan 25, 2025: use of identity function for hashing in a hashmap
- #62 - cmcuza opened on Jan 8, 2025: Automate Publishing to PyPI Using GitHub Actions
- #59 - skejserjensen opened on Dec 23, 2024: Implement Camel (core enhancement)
- #48 - skejserjensen opened on Oct 14, 2024: Implement SHRINK

Alternative: Propose Your Own Topic Idea



- **We are open to additional topic proposals**
 - In the context of data engineering, data management, and machine learning systems
 - If you are passionate about your idea
 - More topics in DAPHNE, SystemDS, TerseTS, or other open-source systems possible, but contributions might be more difficult to get accepted
 - **If you would like to propose your own topic, approach me by email by May 11, 23:59;** in any case, **also fill in the poll regarding the topic selection with your preferred topics from the list above**