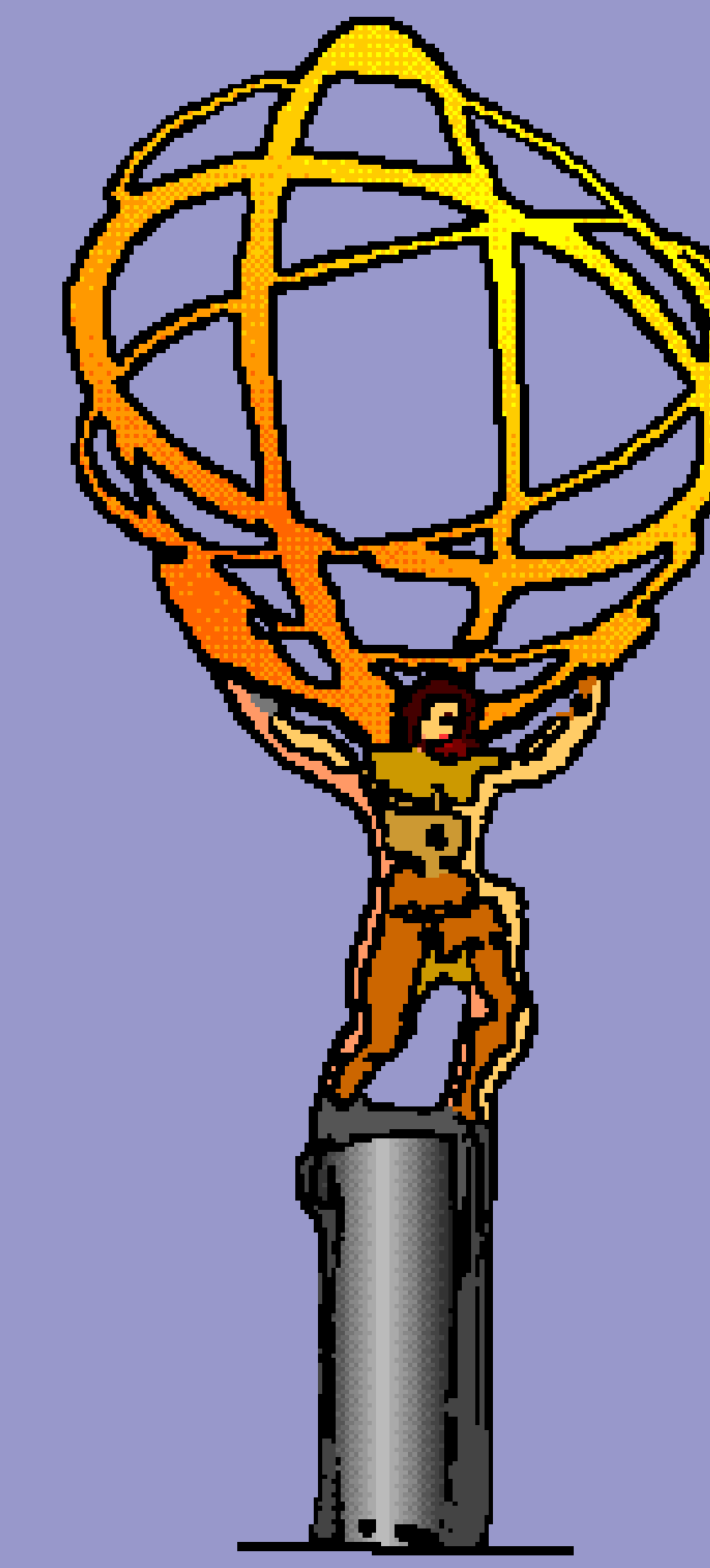


# New Developments in File-based Infrastructure for ATLAS Event Selection

David Malon, Peter van Gemmeren (Argonne National Laboratory),  
Marcin Nowak (Brookhaven National Laboratory)



## 1. Abstract

In ATLAS software, TAGs are event metadata records that can be stored in various technologies, including ROOT files and relational databases. TAGs are used to identify and extract events that satisfy certain selection predicates, which can be coded as SQL-style queries. TAG collection files support in-file metadata to store information describing all events in the collection. Event Selector functionality has been augmented to provide such collection-level metadata to subsequent algorithms. The ATLAS I/O framework has been extended to allow computational processing of TAG attributes to select or reject events without reading the event data.

This capability enables physicists to use more detailed selection criteria than are feasible in an SQL query. For example, the TAGs contain enough information not only to check the number of electrons, but also to calculate their distance to the closest jet—a calculation that would be difficult to express in SQL. Another new development allows ATLAS to write TAGs directly into event data files. This feature can improve performance by supporting advanced event selection capabilities, including computational processing of TAG information, without the need for external TAG file or database access.

## 2. Details

### TAG collection support for in-file metadata

- Since POOL version 2.8, Collections support in-file Metadata.
  - Stored as Key/Value map<string, string>.
  - Recorded in StoreGate.
  - AthenaPOOL fires incident to alert for end TAG file, begin TAG file.

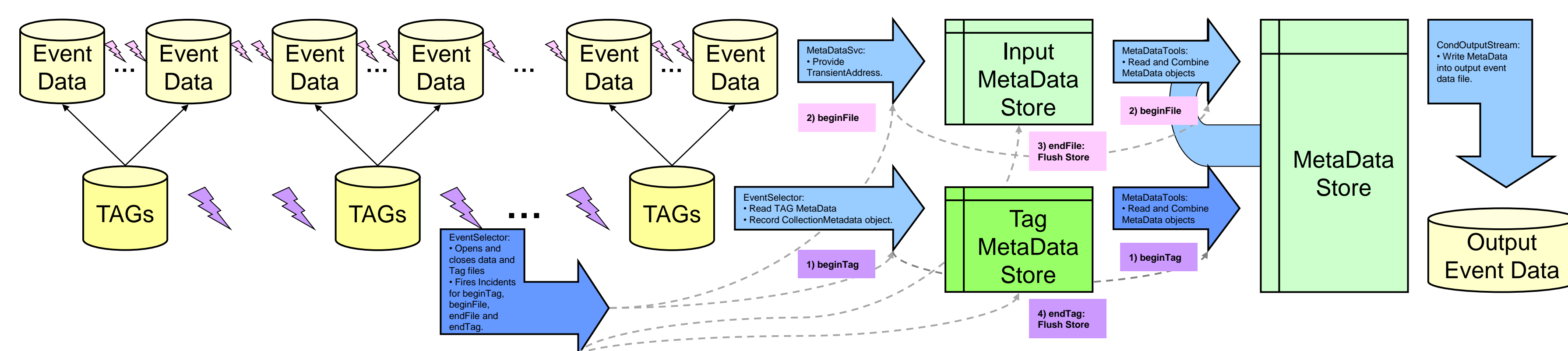


Figure 1: The AthenaPOOL framework has been augmented to support both event file and TAG file metadata.

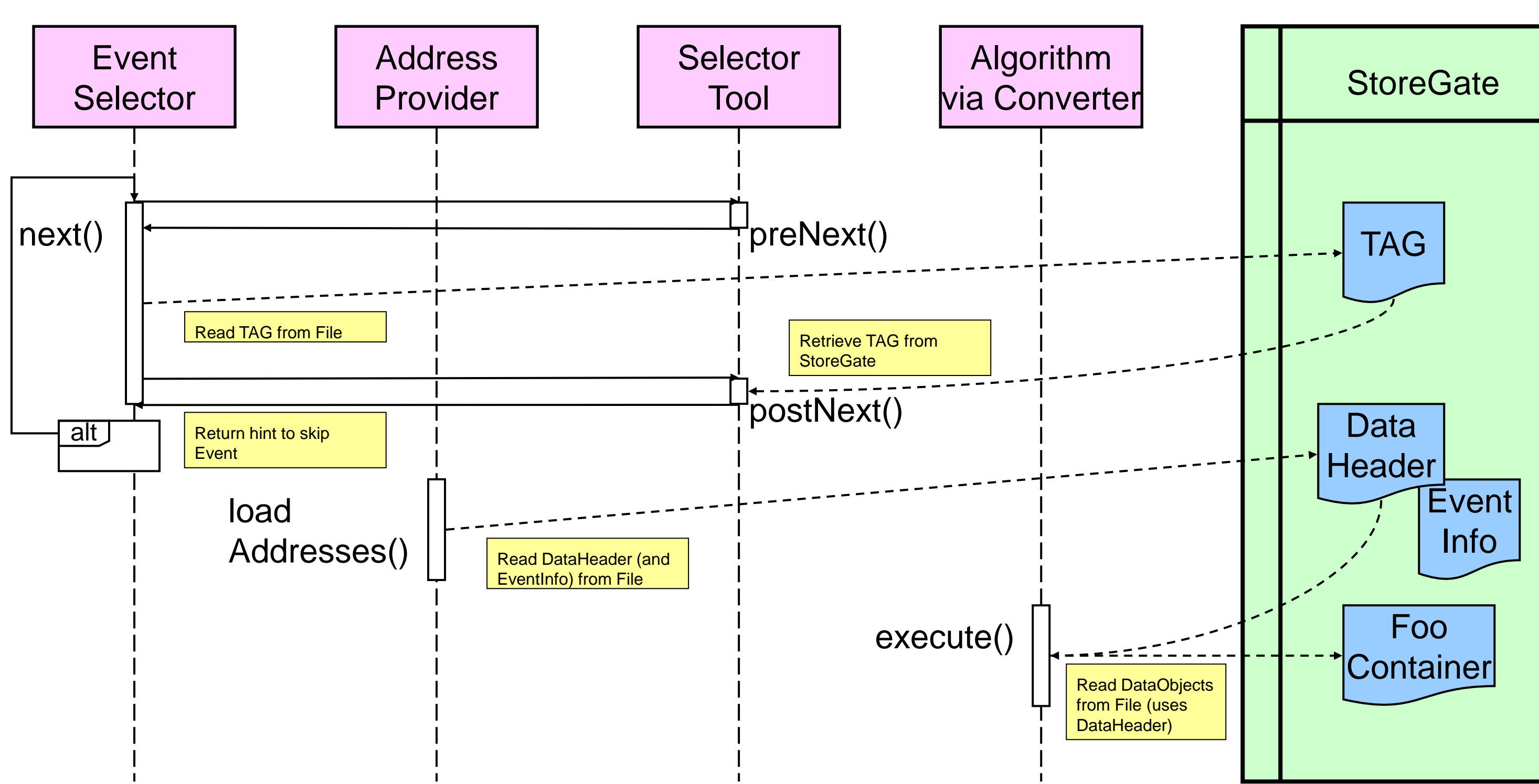


Figure 2: Computational processing of TAG attributes enables physicists to reject events using complex queries without reading the event data

### Computational processing of TAG attributes

- The ATLAS I/O framework has been extended to allow computational processing of TAG attributes to **select/reject** events **without reading the event data** (not even **DataHeader** or **EventInfo**).
- This capability enables physicists to preselect events with more detailed selection criteria than are feasible in an SQL query.
  - For example, the TAGs contain enough information to not only check the number of electrons, but also calculate their **distance to the closest jet**. Iterative and procedural computations are difficult in SQL.
  - Data stored in TAGs suffice for **invariant mass computations**. Combinatorics and SQL queries don't mix well.
- Expect deselecting events based upon TAG content to be **10 times faster** on average than reading the same information from AOD.
  - Given that the size of **TAGs is only ~1KB/event**, we expect substantial improvements for most analyses.
  - Depends upon how many containers would need to be accessed to compute the query.
    - In any case the DataHeader and EventInfo object would be read.
  - Greatest effect for highly selective analysis.

### TAGs within event data files

- Event selection with TAGs is **becoming more popular**.
- It would be nice to be able to apply the **TAG tools directly to AOD**.
  - Allows transparent use of **TAG functionality** (e.g., queries) on **AOD files**.
  - TAGs are **very small** compared to AOD (less than 1%).
  - First TAGs are **produced during AOD merging**.
  - TAGs and event data are both written via **POOL using ROOT I/O**.
- **Issues are mainly technical**:
  - **Synchronization** of multiple writers to the same output file.
    - Can write TAG collections during finalize, after event data are done.
  - **Distinguishability** of Payload and TAG Collection containers
    - Names are configurable
    - ATLAS event selectors are already capable of iteration via either source
- Work in progress.

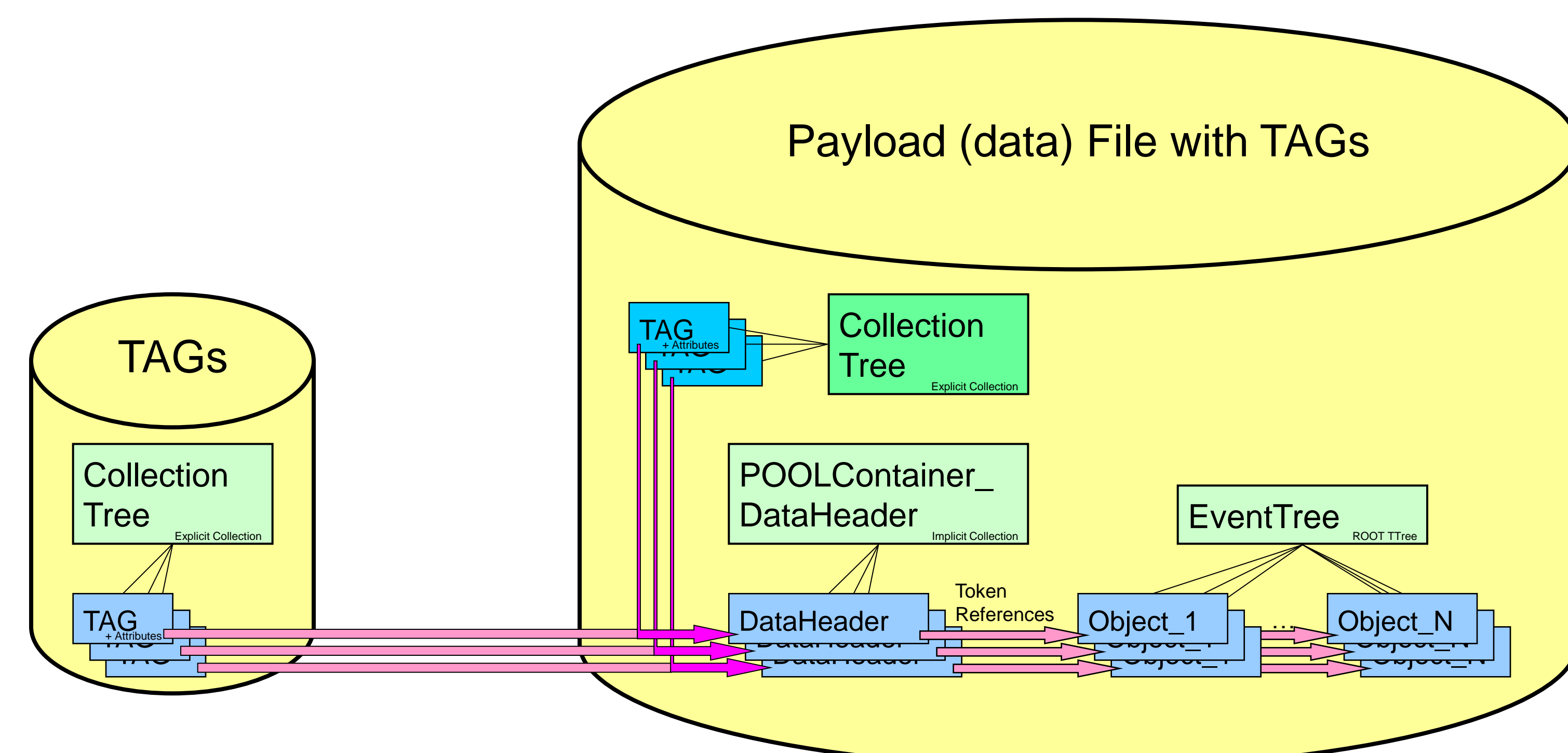


Figure 3: Writing TAGs into event data files allows using TAG event selection capabilities on the data file.