

# PTool: A Light Weight Persistent Object Manager\*

R. L. Grossman, D. Hanley, and X. Qin  
Laboratory for Advanced Computing  
University of Illinois at Chicago

Analyzing, mining and extracting information from data does not require a full featured database, but rather simply the ability to work with persistent data. Advantages of “light weight” data management include: 1) Light weight software tools allow applications to have high performance, low overhead access to persistent data. 2) Light weight software tools can easily be customized and optimized.

We have developed a software tool called PTool that provides high performance, low overhead access to persistent data. PTool is usually used as a library, which when linked to a C++ application provides persistence for instances of C++ classes which are created using an overloaded “new” operator.

Our primary target applications are scientific computing and data mining. For this reason, PTool does not support transactions, back up and recovery, or any of the other functionality usually associated with an object oriented database. Furthermore, our applications primarily involve read only data and generally require very high performance. PTool allows the programmer to specify when they need only read access, allowing it to optimize access time.

The physical design of PTool is based upon the management of contiguous physical collections of memory pages, or *segments*, and physical collections of segments which we call *folios*. There are three basic components to the system: a Persistent Object Manager and a Segment and Folio Manager.

The philosophy underlying the design of PTool is to rely on functions of the operating system, file system and compiler whenever possible, even at the risk of somewhat reduced performance. For example, PTool uses the operating system’s memory management facilities to provide the basic persistence; the file system’s input-output operations to manage the segments and

folios; and the compiler’s bit operations for addressing. PTool is designed for:

*Scalability.* PTool was designed to scale to the Terabyte range and beyond. For example, it has been used to manage high energy physics data at Fermi National Accelerator Laboratory which was distributed over one hundred Gigabytes of spinning disk and a Terabyte of 8mm tape. To achieve this scalability, PTool supports a physical hierarchy consisting of segments and folios, with separate segment and folio managers. This simple idea is commonly used in hierarchical storage systems, but much less common in data management systems. Simply put, a Terabyte of data organized into segments alone would consist of too many segments to manage efficiently.

*Flexibility.* The interface between the different components make use of virtual functions and an “envelope-letter” coding paradigm. Several Segment Managers have been developed and used: for example, there are Segment Managers which interface to 8mm tape, which manage segments in local area ATM clusters, and which compress segments before sending them to improve performance for wide area clusters. Different stores can use different Segment Managers: an attribute of the store specifies the Segment Manager which was used to populate the store. This keeps each of the Segment Managers small and light weight and allows applications to select, at run time, the appropriate Segment Manager for working with a particular store.

*Adaptability.* By keeping PTool “light-weight,” it is feasible for projects to develop custom versions of PTool suited for specific architectures or applications. For example, the Fermi Lab application mentioned above developed specialized Segment Managers which exploit the high performance communication protocols of an IBM SP-2 switch.

---

\*This research was supported in part by NASA grant NAG2-513, DOE grant DE-FG02-92ER25133, and NSF grants IRI 9224605 and CDA-9303433. For more information, contact Robert Grossman, Laboratory for Advanced Computing, m/c 249, University of Illinois at Chicago, 851 South Morgan Street, Chicago, IL 60607, grossman@uic.edu.