

# VIDEOS

## Starburst II: The Extender Strikes Back!

Guy M. Lohman, George Lapis, Tobin Lehman, Rakesh Agrawal, Roberta Cochrane  
John McPherson, C. Mohan, Hamid Pirahesh, Jennifer Widom

IBM Almaden Research Center

San Jose, CA 95120

{lohman, lapis, toby, ragrawal, johnm, mohan, pirahesh, widom}@ibm.com, bobbie@cs.umd.edu

Starburst is an extensible relational database system prototype developed at IBM's Almaden Research Center. The goal throughout its development has been to build a *complete relational database system kernel*, engineered throughout with the *infrastructure* for *generic* extensions, rather than specific extensions for a single application area not previously supportable by relational technology. Currently, Starburst consists of over 300,000 lines of C (and some C++) source code, and executes an extended form of the SQL Data Definition language and the SQL SELECT, INSERT, and DELETE commands, including joins on any number of tables. We demonstrate it on IBM's first RISC-technology workstation, the RT PC, under the AIX operating system (based on Unix V<sup>1</sup>).

Our 1990 SIGMOD video demonstrated several aspects of Starburst that facilitated adding extensions to it. This demonstration highlights four extensions implemented in Starburst:

1. The *IMS attachment* automatically maintains parent-child pointers and clustering to speed up joins that are frequently performed between a parent and its children, and to enforce a limited form of referential integrity. IMS attachments allow Starburst to quickly build hierarchical objects from data that is stored relationally in tables.
2. *Very Long Fields* permit objects up to 1.5 gigabytes in length (such as images or voice) to be managed by Starburst as though they are individual objects in a column. Since Starburst's Long Field Manager directly manages these objects on raw disk, it speeds access by ensuring contiguous disk areas and avoiding AIX buffering.
3. *User-Defined Production Rules* are triggered by user-specified sets of changes to tables at the end of a transaction, test conditions that are expressed as any SQL query and may reference the changes that triggered them, and if satisfied, invoke actions that may include any SQL Data Definition or Data Manipulation command.
4. *Alert Object-Oriented Triggers* are triggered by the invocation of any user-defined operation (or *method*) on any object (including views) at any time, including multi-table operations and operations that span transactions. These rules are defined as SQL views, which may return data as well as triggering actions, and may be nested in the usual way (i.e., exhibit the *closure property*).

Many other extensions of Starburst, such as our EXTended Normal Form (XNF) for defining complex objects as views, could not be included in this demonstration.

---

<sup>1</sup> Unix is a trademark of AT&T