

Extended File System Metadata Management with Relational Databases

Michael McThrow

Carlos Maltzahn

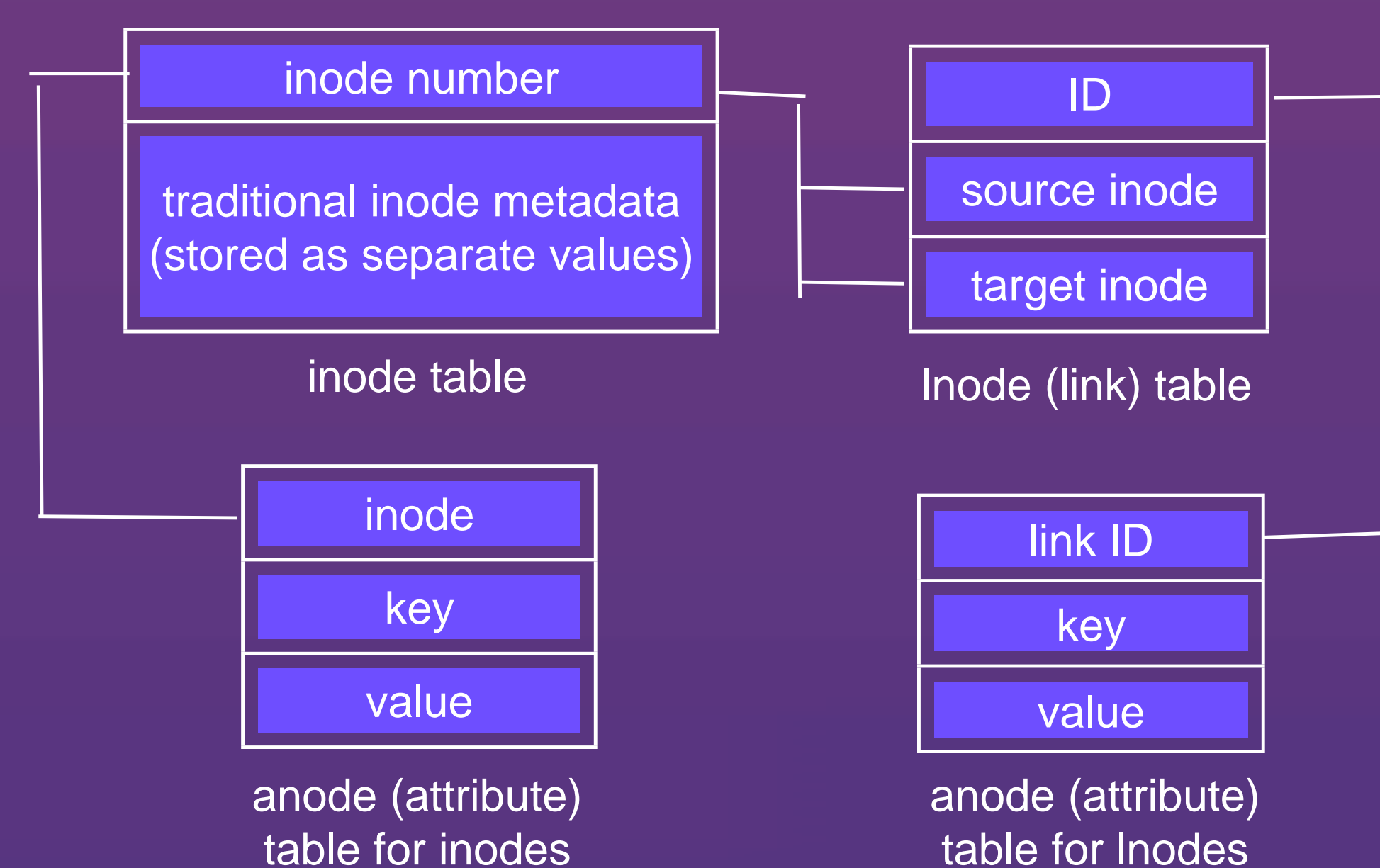
Scott Brandt

Storage Systems Research Center at the University of California, Santa Cruz

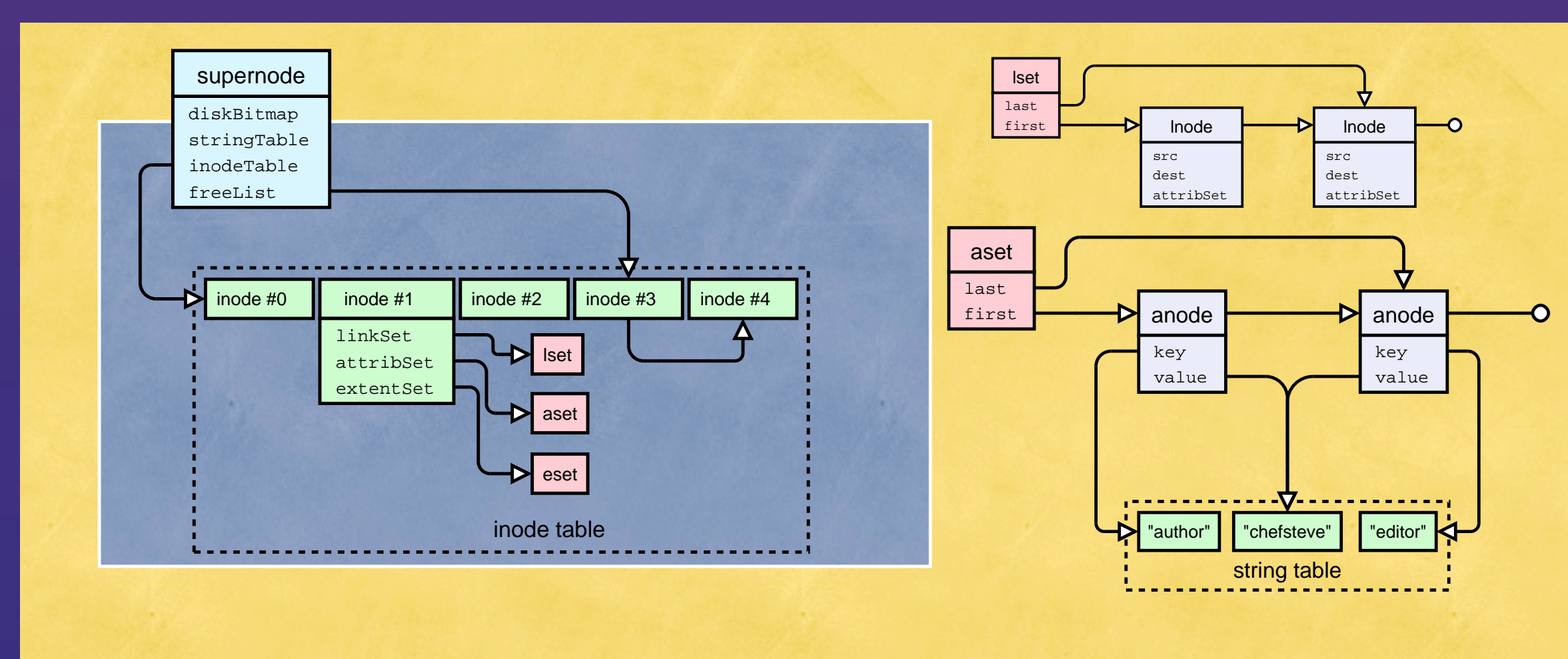
Extended Metadata

- File attributes (key/value pairs).
- Relational links between files
- Relational link attributes (key/value pairs)

Here: Relational Database



Before: Native Data Structures (LiFS)



Motivation

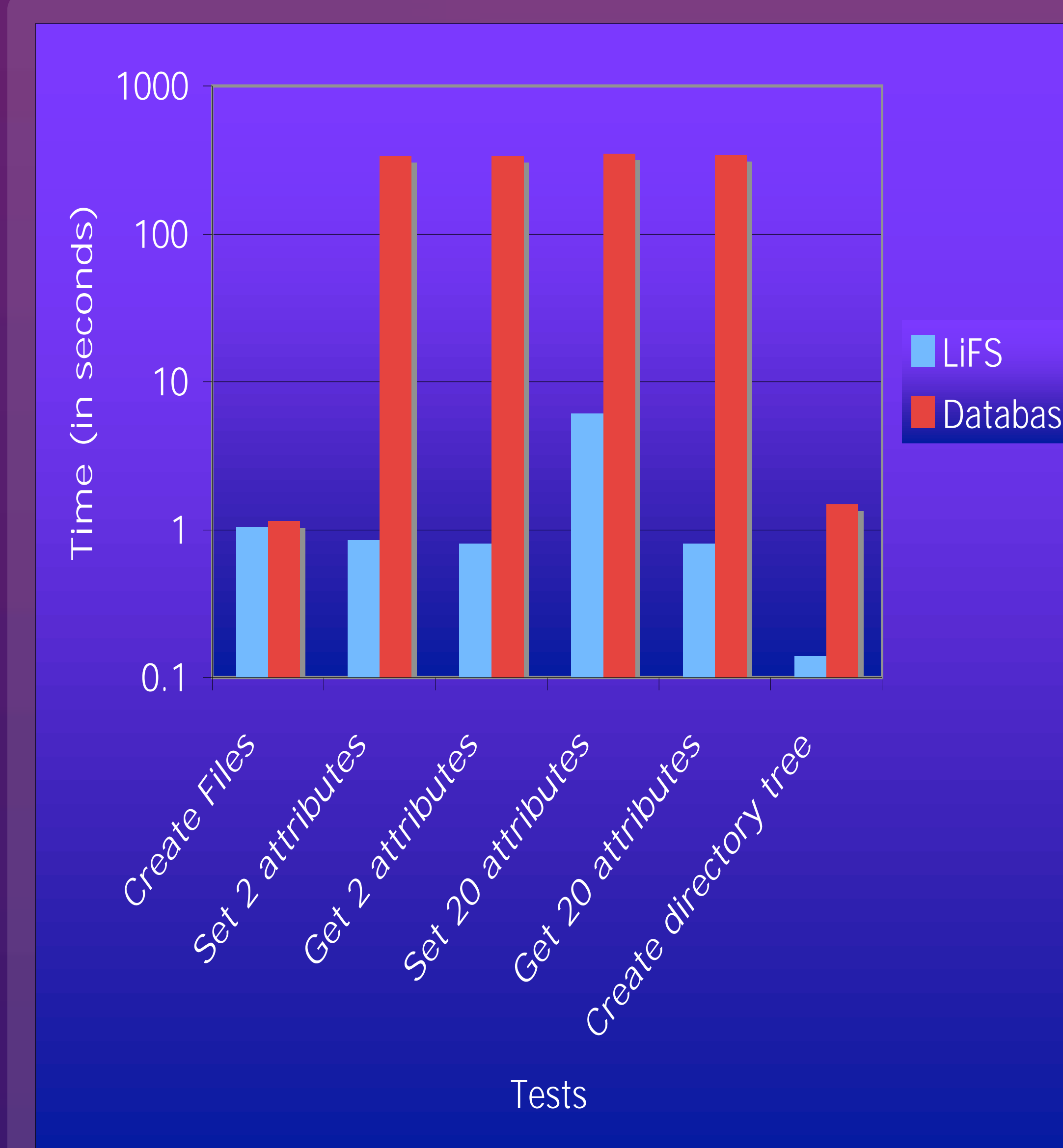
- Modern file systems need to handle extended metadata.
- Databases are an established technology for storing and organizing data.
- Are databases a better way to manage extended metadata than native file system data structures?

Evaluation

- Comparison to Linking File System (LiFS) published performance numbers [Ames et al. 2006]
- In-memory database (SQLite)
- Timing (and scaling) of
 - directory creation
 - zero-length file creation
 - file attribute creation and deletion
 - link creation and deletion
 - link attribute creation

Future Work

- We will optimize our database by researching alternate schemata.
- We will also implement an actual file system
- We will compare the performance of the file system to other file systems.



Results

- Based on a directory tree with 5 sub-directories, 4 files per directory (excluding the root directory), and a depth of 5.
- File creation and directory tree creation times are competitive with the native data structures.
- Attribute creation and retrieval are not.
- **More results in the paper.**

This work was completed as part of UCSC's SURF-IT summer undergraduate research program, an NSF CISE REU Site. This material is based upon work supported by the National Science Foundation under Grant No. CCF-0552688.