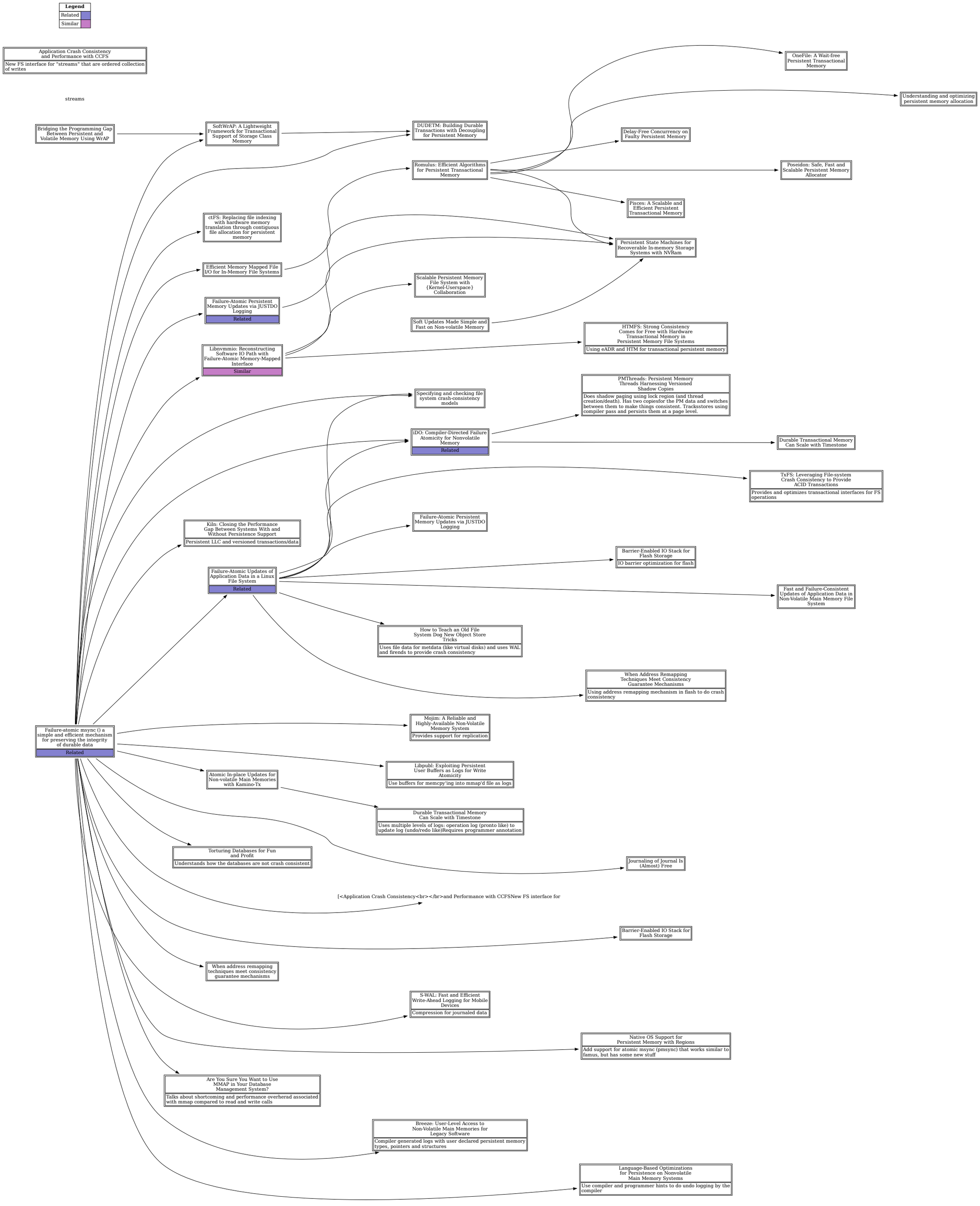
Legend
Related
Similar
Application Crash Consistency and Performance with CCFS
New FS interface for "streams" that are ordered collection of writes
streams
Bridging the Programming Gap Between Persistent and Volatile Memory Using WrAP
SoftWrAP: A Lightweight Framework for Transactional Support of Storage Class Memory
DUDETM: Building Durable Transactions with Decoupling for Persistent Memory
OneFile: A Wait-free Persistent Transactional Memory
Understanding and optimizing persistent memory allocation
Delay-Free Concurrency on Faulty Persistent Memory
Romulus: Efficient Algorithms for Persistent Transactional Memory
Poseidon: Safe, Fast and Scalable Persistent Memory Allocator
Pisces: A Scalable and Efficient Persistent Transactional Memory
ctFS: Replacing file indexing with hardware memory translation through contiguous file allocation for persistent memory
Persistent State Machines for Recoverable In-memory Storage Systems with NVRam
Efficient Memory Mapped File I/O for In-Memory File Systems
Scalable Persistent Memory File System with {Kernel-Userspace} Collaboration
Failure-Atomic Persistent Memory Updates via JUSTDO Logging
Related
Soft Updates Made Simple and Fast on Non-volatile Memory
HTMFS: Strong Consistency Comes for Free with Hardware Transactional Memory in Persistent Memory File Systems
Using eADR and HTM for transactional persistent memory
Libnvmmio: Reconstructing Software IO Path with Failure-Atomic Memory-Mapped Interface
Similar
PMThreads: Persistent Memory Threads Harnessing Versioned Shadow Copies
Does shadow paging using lock region (and thread creation/death). Has two copiesfor the PM data and switches between them to make things consistent. Tracksstores using compiler pass and persists them at a page level.
Specifying and checking file system crash-consistency models
iDO: Compiler-Directed Failure Atomicity for Nonvolatile Memory
Related
Durable Transactional Memory Can Scale with Timestone
TxFS: Leveraging File-system Crash Consistency to Provide ACID Transactions
Provides and optimizes transactional interfaces for FS operations
Kiln: Closing the Performance Gap Between Systems With and Without Persistence Support
Persistent LLC and versioned transactions/data
Failure-Atomic Persistent Memory Updates via JUSTDO Logging
Failure-Atomic Updates of Application Data in a Linux File System
Related
Barrier-Enabled IO Stack for Flash Storage
IO barrier optimization for flash
Fast and Failure-Consistent Updates of Application Data in Non-Volatile Main Memory File System
How to Teach an Old File System Dog New Object Store Tricks
Uses file data for metdata (like virtual disks) and uses WAL and firends to provide crash consistency
When Address Remapping Techniques Meet Consistency Guarantee Mechanisms
Using address remapping mechanism in flash to do crash consistency
Failure-atomic msync () a simple and efficient mechanism for preserving the integrity of durable data
Related
Mojim: A Reliable and Highly-Available Non-Volatile Memory System
Provides support for replication
Libpubl: Exploiting Persistent User Buffers as Logs for Write Atomicity
Use buffers for memcpy'ing into mmap'd file as logs
Atomic In-place Updates for Non-volatile Main Memories with Kamino-Tx
Durable Transactional Memory Can Scale with Timestone
Uses multiple levels of logs: operation log (pronto like) to update log (undo/redo like)Requires programmer annotation
Torturing Databases for Fun and Profit
Understands how the databases are not crash consistent
Journaling of Journal Is (Almost) Free
[<Application Crash Consistency
</br>and Performance with CCFSNew FS interface for
Barrier-Enabled IO Stack for Flash Storage
When address remapping techniques meet consistency guarantee mechanisms
S-WAL: Fast and Efficient Write-Ahead Logging for Mobile Devices
Compression for journaled data
Native OS Support for Persistent Memory with Regions
Add support for atomic msync (pmsync) that works similar to famus, but has some new stuff
Are You Sure You Want to Use MMAP in Your Database Management System?
Talks about shortcoming and performance overherad associated with mmap compared to read and write calls
Breeze: User-Level Access to Non-Volatile Main Memories for Legacy Software
Compiler generated logs with user declared persistent memory types, pointers and structures
Language-Based Optimizations for Persistence on Nonvolatile Main Memory Systems
Use compiler and programmer hints to do undo logging by the compiler