# Don’t Give Up on Distributed File Systems

Jeremy Stribling, Emil Sit,
Frans Kaashoek, Jinyang Li, and Robert Morris

*MIT CSAIL and NYU*

# Reinventing the Storage Wheel

- New apps tend to use new storage layers
- Examples:

Over Cite
CORAL
CoDeeN
oceanstore
PLANETLAB
BLAST

- Can we invent this layer once?

# What About a File System?

- A FS enables quick-prototyping for apps
  - A familiar interface
  - Language-independent usage model
  - Hierarchical namespace useful for apps
  - Write distributed apps in shell scripts

```
if [ –f /fs/cwc/$URL ]; then
    if notexpired /fs/cwc/$URL; then
        cat /fs/cwc/$URL
        exit
    fi
fi
wget $URL –O – | tee /fs/cwc/$URL
```

# Why Won’t That Work Today?

- Needs of distributed apps:
  - Control over consistency and delays
  - Efficient data sharing between peers
- Current systems focus on FS transparency
  - Hide faults with long timeouts
  - Centralized file servers

# Example: Cooperative Web Cache

- Would rather fail and refetch than wait
- Perfect consistency isn’t crucial
- Avoid hotspots

```
if [ –f /fs/cwc/$URL ]; then                  ←
    if notexpired /fs/cwc/$URL; then          ←
        cat /fs/cwc/$URL                      ←
        exit
    fi
fi
wget $URL –O – | tee /fs/cwc/$URL             ←
```

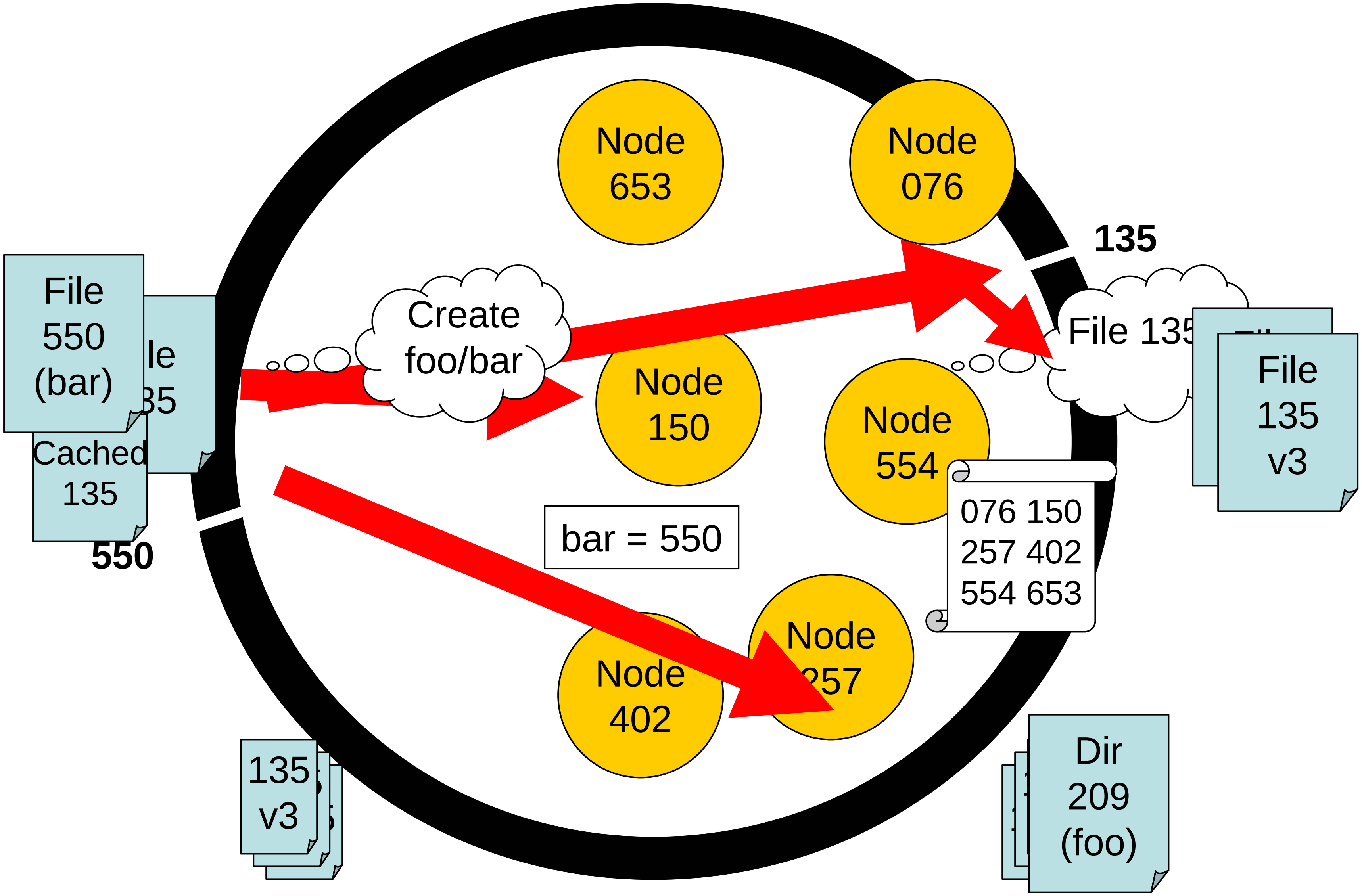
# Our Proposal: WheelFS

- A distributed wide-area FS to simplify apps
- Main contributions:

  1) Give apps control with *semantic cues*

  2) Provide good performance according to *Read Globally, Write Locally*

# Basic Design: Reading and Writing

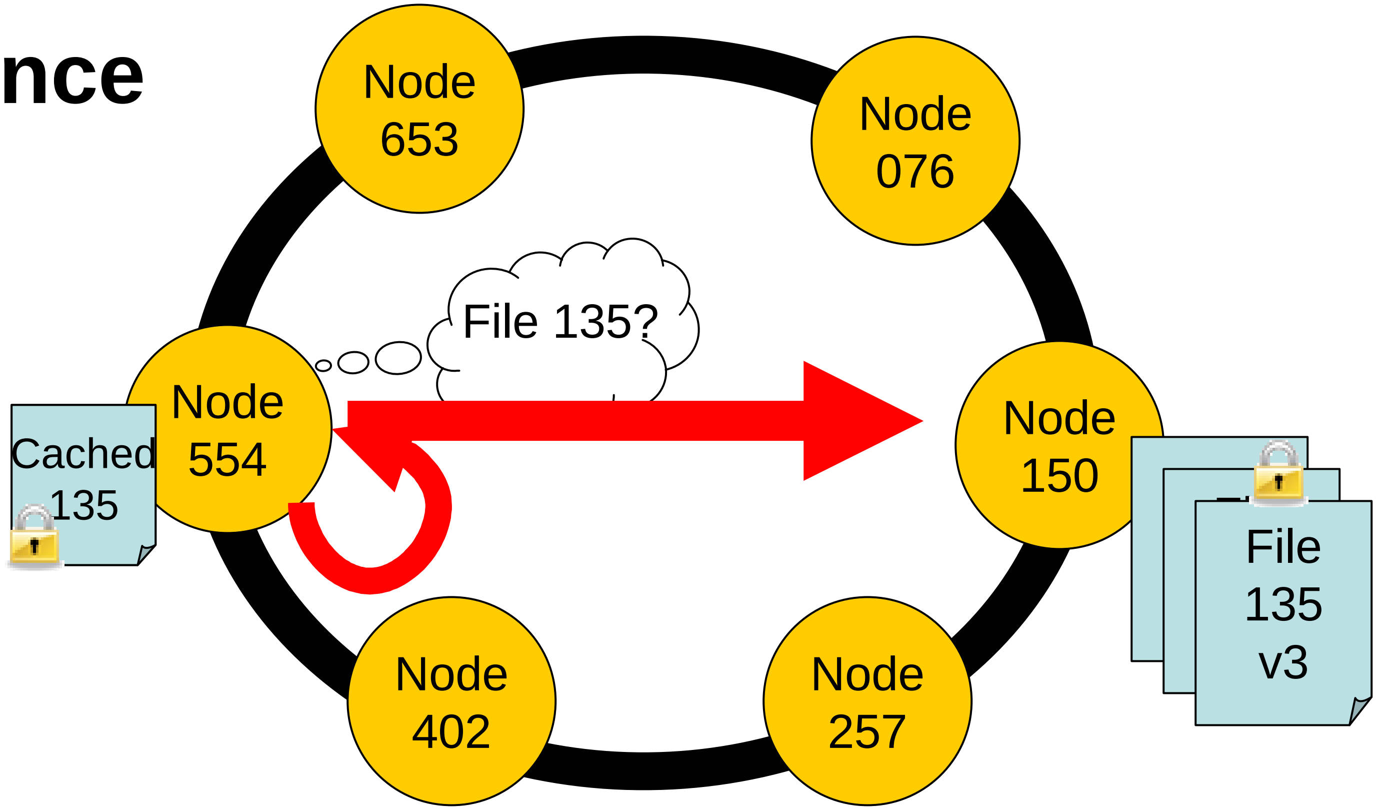


# Explicit Semantic Cues

- Allow direct control over system behavior
- Meta-data that attach to files, dirs, or refs
- Apply recursively down dir tree
- Possible impl: intra-path component
  - *[/wfs/cwc/.](/wfs/cwc/.)**cue**/foo/bar*

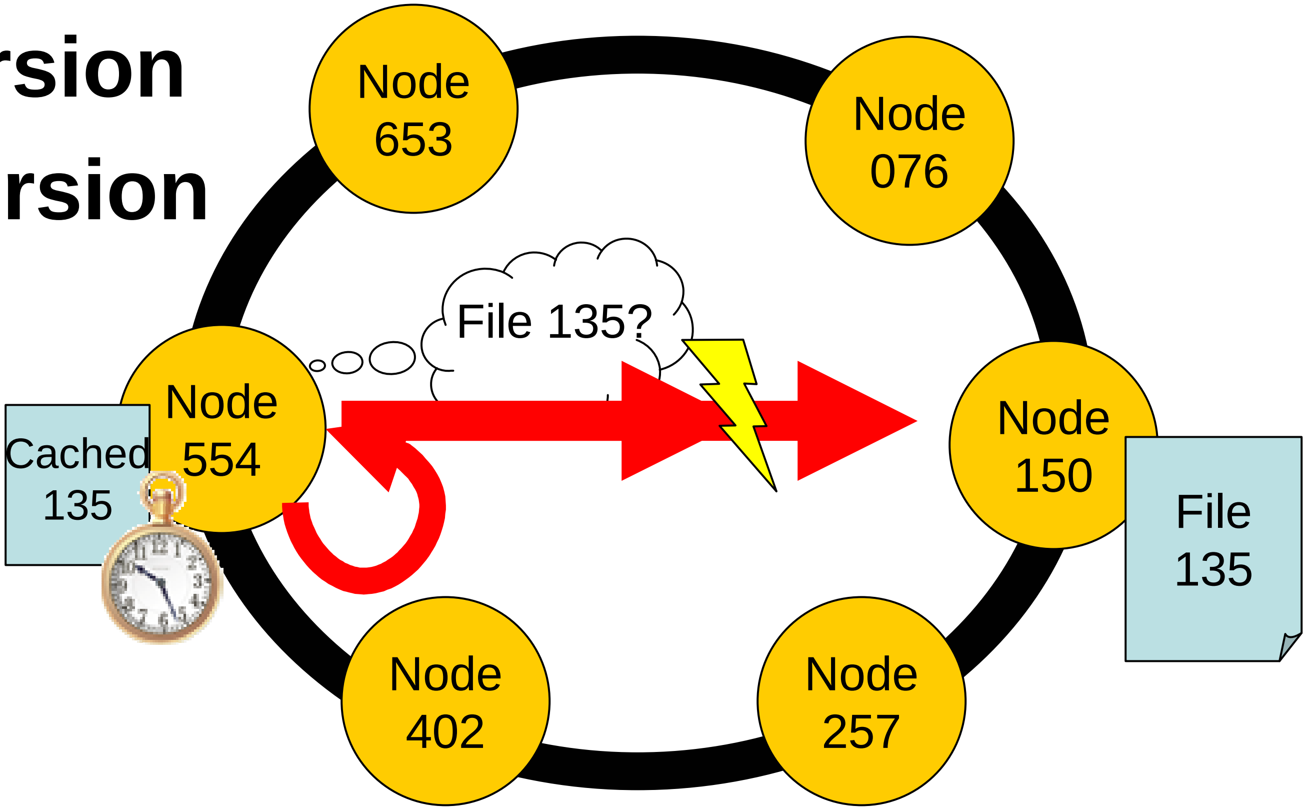
# Semantic Cues: Writability

- Applies to files
- **WriteMany** (default)
- **WriteOnce**



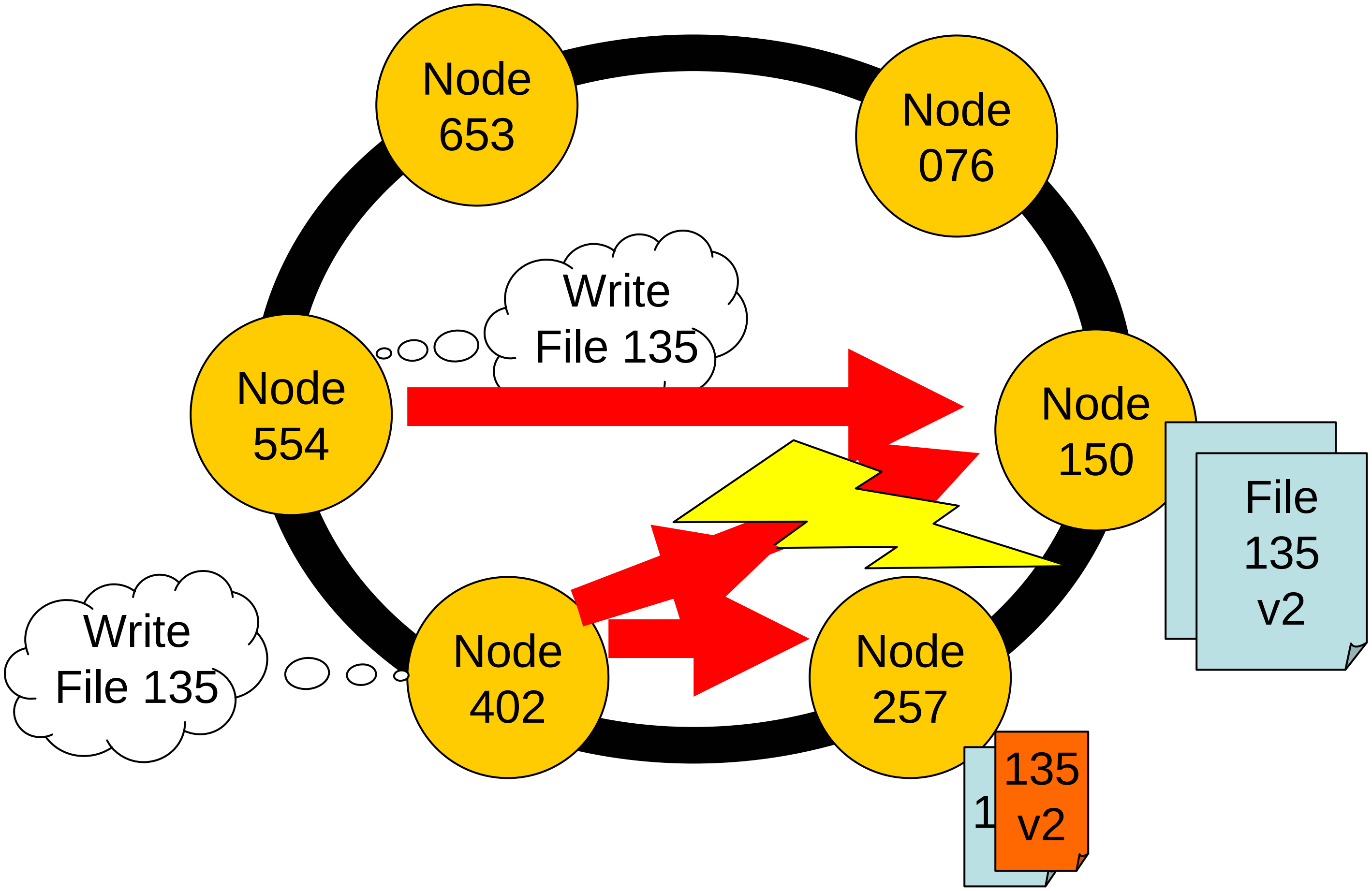
# Semantic Cues: Freshness

- Applies to file references
- **LatestVersion** (default)
- **AnyVersion**
- **BestVersion**



# Semantic Cues: Write Consistency

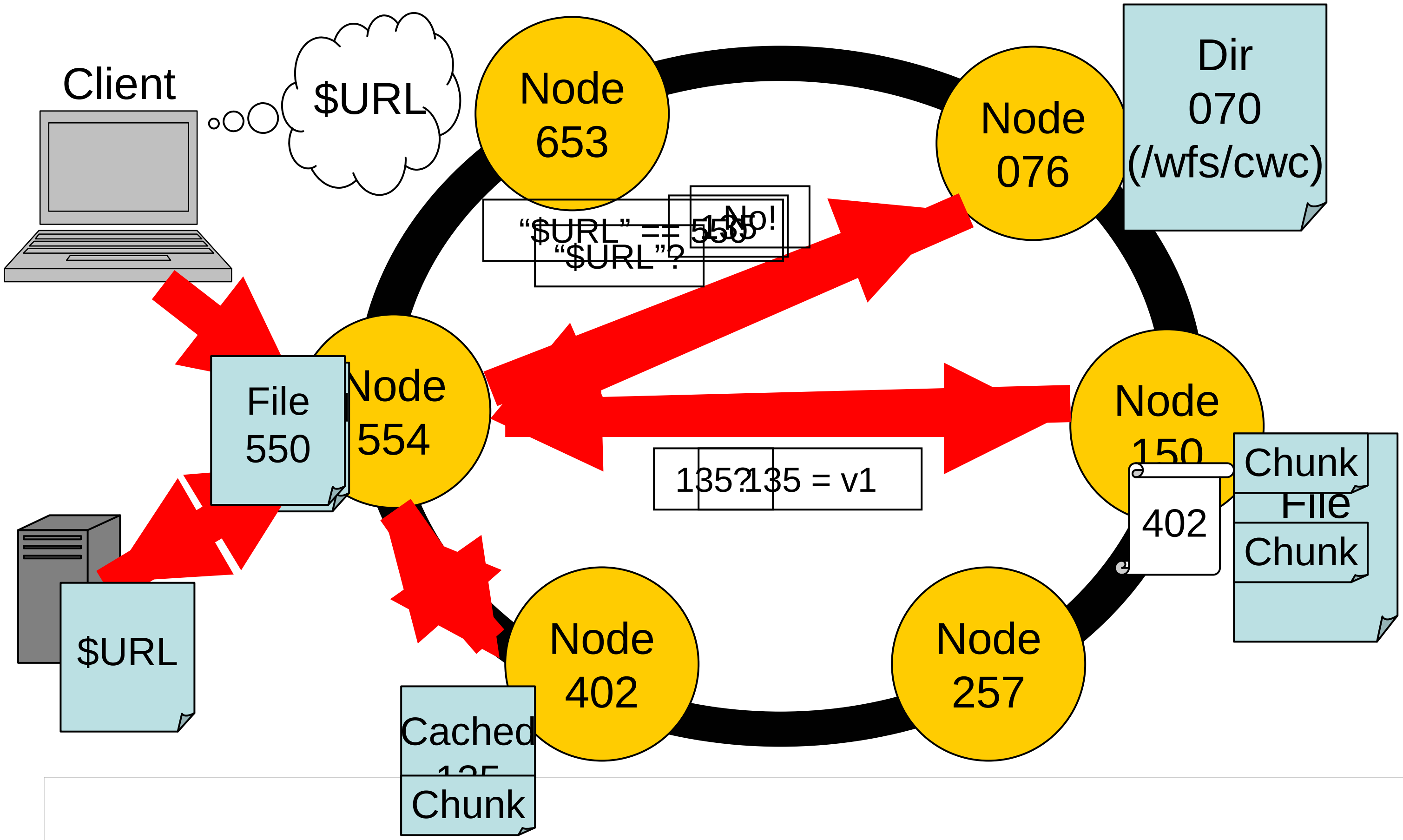
- Applies to files or directories
- **Strict** (default)
- **Lax**



# Example: Cooperative Web Cache

- Reading an older version is ok:
  - `cat /wfs/cwc/.bestversion,maxtime=250/foo`
- Writing conflicting versions is ok:
  - `wget http://foo > /wfs/cwc/.lax,writemany/foo`

# Example: Cooperative Web Cache



en

# Discussion

- Current set of cues enough for many apps
  - All-sites-pings
  - Grid computations
  - OverCite
- Stuff we swept under the rug:
  - Security
  - Atomic renames across dirs
  - Storage load-balancing
  - Unreferenced files

# Related Work

- Every FS paper ever written
- Specifically:
  - Cluster FS: Farsite, GFS, xFS, Ceph
  - Wide-area FS: JetFile, CFS, Shark
  - Grid: LegionFS, GridFTP, IBP
  - POSIX I/O High Performance Computing Extensions

# Conclusion

- WheelFS: distributed storage layer for newly-written applications
- Control through explicit semantic cues
- Performance by reading globally and writing locally