

User-level Checkpointing Through Exportable Kernel State

Patrick Tullmann, Jay Lepreau, Bryan Ford, Mike Hibler

October 28, 1996

Flux Project

University of Utah

<http://www.cs.utah.edu/projects/flux>

Key Points

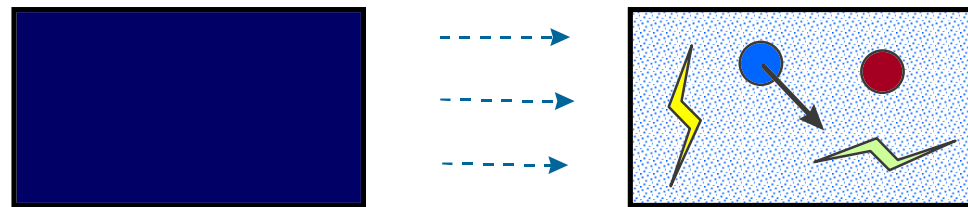
- ❑ Always exportable/settable kernel state:
 - Allows interesting user-level OS services
 - Has not been provided before
 - Can be done: Fluke does it
- ❑ Flexible checkpointing is an example service

Visible Kernel State

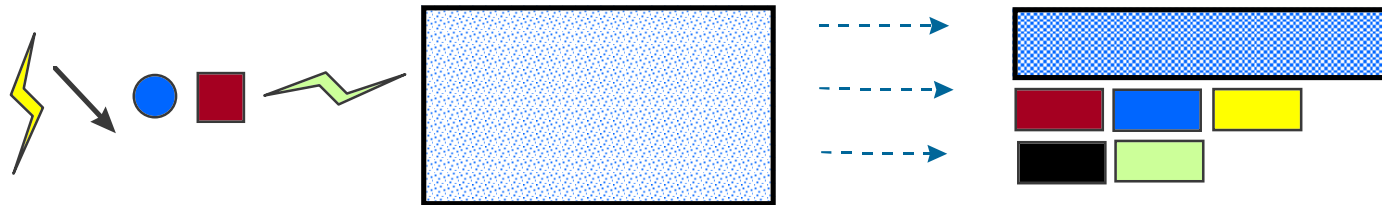
- ❑ State exported from or imported to kernel
 - Encapsulated in kernel objects, e.g. threads, ports, ...
- ❑ State must *always* be available
- ❑ Enables user-level services
 - Process migration, distributed memory, debugging, memory management
 - Checkpointing

Checkpointing: Key Issues

- Find the associated kernel objects



- Extract state from those objects



Checkpointing: The Rest

- ❑ Pickle the object state
- ❑ Save it
- ❑ Bring it back and re-create the target

Sample Kernel Object State

□ Thread State

- Registers
- IPC state
- Exception handlers
- References:
 - Scheduler thread
 - IPC Client
 - IPC Server
 - Address space

□ Memory Map State

- Offset in source
- Size
- Start address
- Protection
- References:
 - Source space
 - Address space

Hasn't This Been Done?

❑ Amoeba

- Restrictions on when export is feasible, e.g., not during IPC operations

❑ Mach

- Restrictions on when export is feasible, e.g., not during long-term IPC operations

❑ V++ Cache Kernel

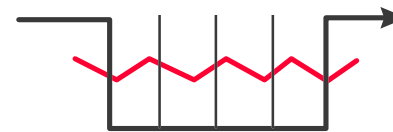
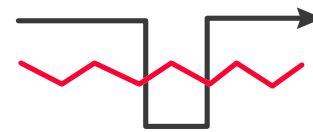
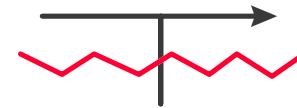
- Strict ordering restrictions on export, e.g., all “dependent” objects *must* be exported

How Fluke Does It

- All necessary kernel state is in objects
 - All implicit kernel state is re-createable
- All operations on objects *appear* atomic
 - Hides implicit kernel state
 - Simplifies object state

Atomic & Restartable Operations

- ❑ Atomic operations
 - avoid intermediate state
- ❑ “Short” operations
 - undo changes if interrupted
- ❑ “Long” operations
 - break into valid intermediate states



Status

- ❑ Implemented on x86 PCs
- ❑ Checkpointer checkpoints and restores arbitrary subsystems
- ❑ Kernel hosts many user-level services
 - Virtual memory, debugging, process management
 - Many Unix utilities including GCC
- ❑ Expected release within 2 months

Conclusion

- Cleanly Visible Kernel State:
 - Enables user-level OS services
 - Is tricky to provide
 - Is feasible - Fluke does it

More papers on Fluke at OSDI'96