Advanced Topics in Computer Systems, CS262B
Prof Eric A. Brewer

MACEDON

April 20, 2004

I. Background

Distributed systems:

- old model: RPC and/or distributed objects
- new model: cooperating finite-state machines
- claim: overlay networks, DHTs, etc. can be built using an FSM framework

II. Finite-State Machines

State machine for each node

- big states: joining, leaving, init, etc.
- state variables, e.g. neighbor list, timers
- atomic transitions: read/write locks
  - control packets = write
  - data packets = read only

Three kind of transitions:

- API calls: before_state(s) API <api_name> [locking read;] { <actions> }
- message reception: before_state(s) recv|forward <message_name> { <actions> }
- timer events: before_state(s) timer <timer_name> [locking read;] { <actions> }

Actions: work done on a transition

- actions occur on transitions (not in states)
- therefore locking on transitions is typically sufficient
- examples:
  - schedule timer (e.g. timeout)
  - transmit message
  - change state
  - modify state variables

Note that FSM is hard to check statically, uses dynamically linked handlers

III. Auxiliary Support

Basic API:

- init
- forward: route message to next hop
- deliver: arrival at last hop (?)
- notify: upcall about network changes (or lower layers in general)
- groups: create/join/leave
- route/multicast/anycast

Libraries:

- SHA and crypto functions, e.g. MD5, public keys
- basic data structures: hash tables, Bloom filters
- locks
- tracing/logging support
- neighbor lists (e.g. pick a random neighbor)

IV. Code generation

Generates C++ for PlanetLab or ModelNet or NS

input code size is very small (big win): 100-600 lines of FSM code

V. Validation

very important: need to know that generated code is comparable with the real version

Validation done on ModelNet (rather than PlanetLab)

Decent validation for NICE and Chord (app-level multicast and DHT respectively)

Pastry results: Macedon has better performance than FreePastry (probably due to Java)

- implies there is no real performance penalty for the higher-level of abstraction

Also used to explore parameter settings for SplitStream