Supporting Mobility in LSAM

Ted Faber
Wei Yue
Information Sciences Institute

LSAM

Goal

- provide distributed data interactively

Techniques

- enhanced caching and prefetching

Features:

- hierarchical cache
- self-configuring hierarchy
- variable caching strategies
- multicast prefetching
- mobility support

Motivation

We are living in a mobile world

Users move

- Disaster relief
- Researchers
- Military

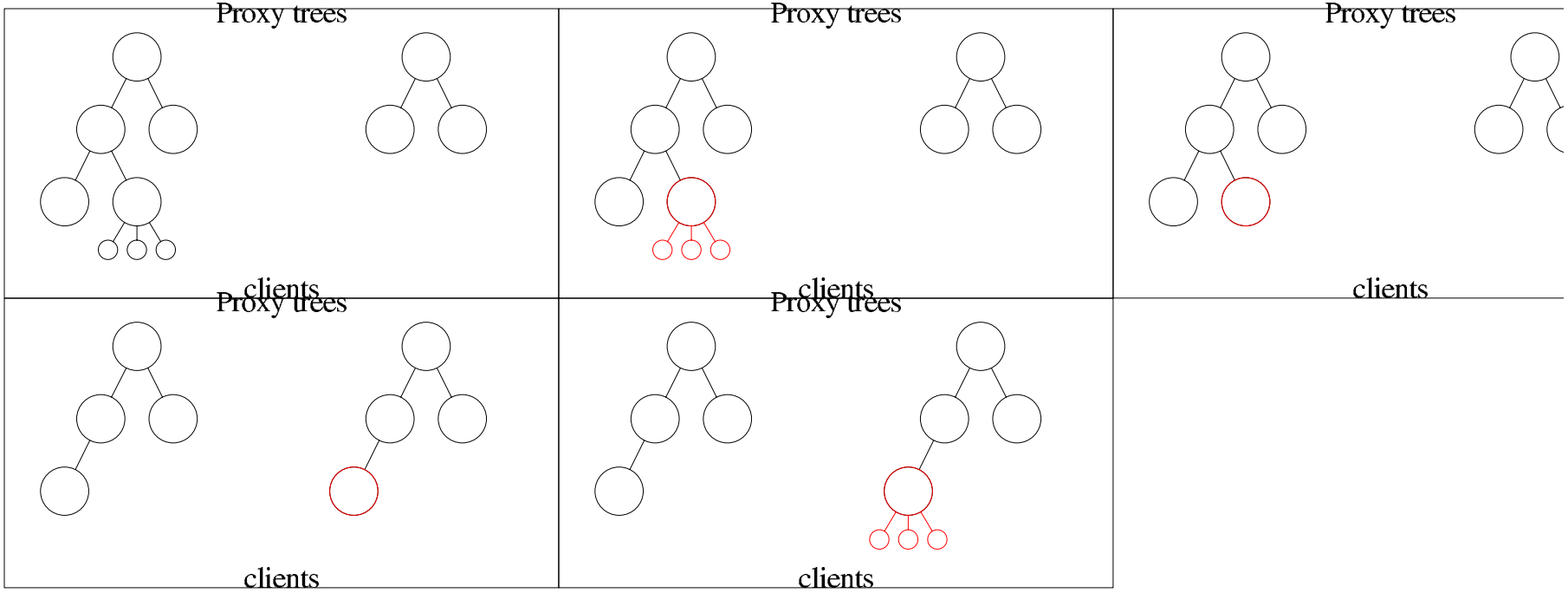
LSAM should move with them

Mobility Model

Unplug and Go

Relocation, not continuous service

Multi-user proxies



Outline

LSAM Overview

Mobility Architecture

- Mobility Servers
- Interface
- Resource Location
- Proxy Migration

Conclusions

Future Work

LSAM Architecture 101

Fast access to information

Hierarchical Cache

- provides natural aggregation points

Multicast Prefetching

- distributes info to a community of interest

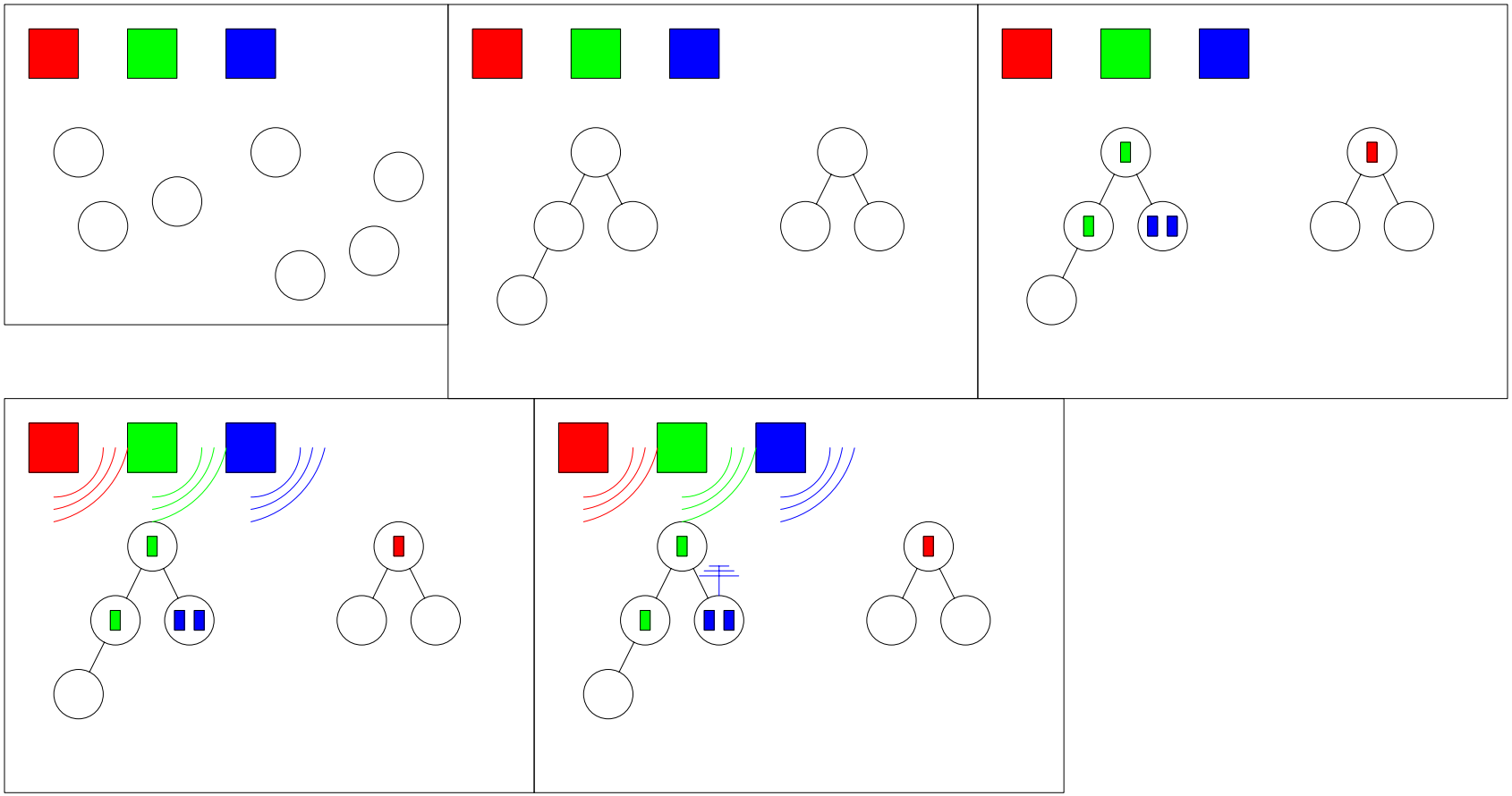
Request Routing

- organizes the hierarchy to follow net structure

Replica Selection

- chooses the most responsive service point

Learning LSAM Visually



Elements of LSAM Mobility

How to get from here to there

Mobility Servers

- proxies that will accept migrating proxies

User Interface

- natural language geographical specification

Resource Location

- find a close mobility server
- integrate network topology information

Proxy Migration

- compactly represent the proxy state
- move state to mobility server

Mobility Servers

Proxy Motels

Home for moving proxies

- sufficient resources to host other proxies
- run proxy transfer protocols

Indexed by Geography Servers

- location included in DNS
- location can be imprecise

User Interface

I don’t want to buy a GPS

Initiating a move

- User describe destination in English
- LSAM selects close mobile servers
- Proxy encapsulates itself and moves

Client arrival

- Clients pick best proxy
- Proxy may migrate to client machine

Resource Location

The Players

Geometry Servers

The Routing Hierarchy

Replica Selection

Geometry Servers

Finding a needle in a haystack

Databases

- Mobility Server locations
- Region names

Query domain

- "Find mobility servers in Washington, D.C."

Data structures are decentralizable

Geometry Server Status

Search routines are implemented

- Find all machines in NYC

Performance

- initialization < 1 minute
  - >15,000 locations
  - > 180,000 machines
- search times are interactive
- untuned implementation

Picking the right server

Geography vs. Topology

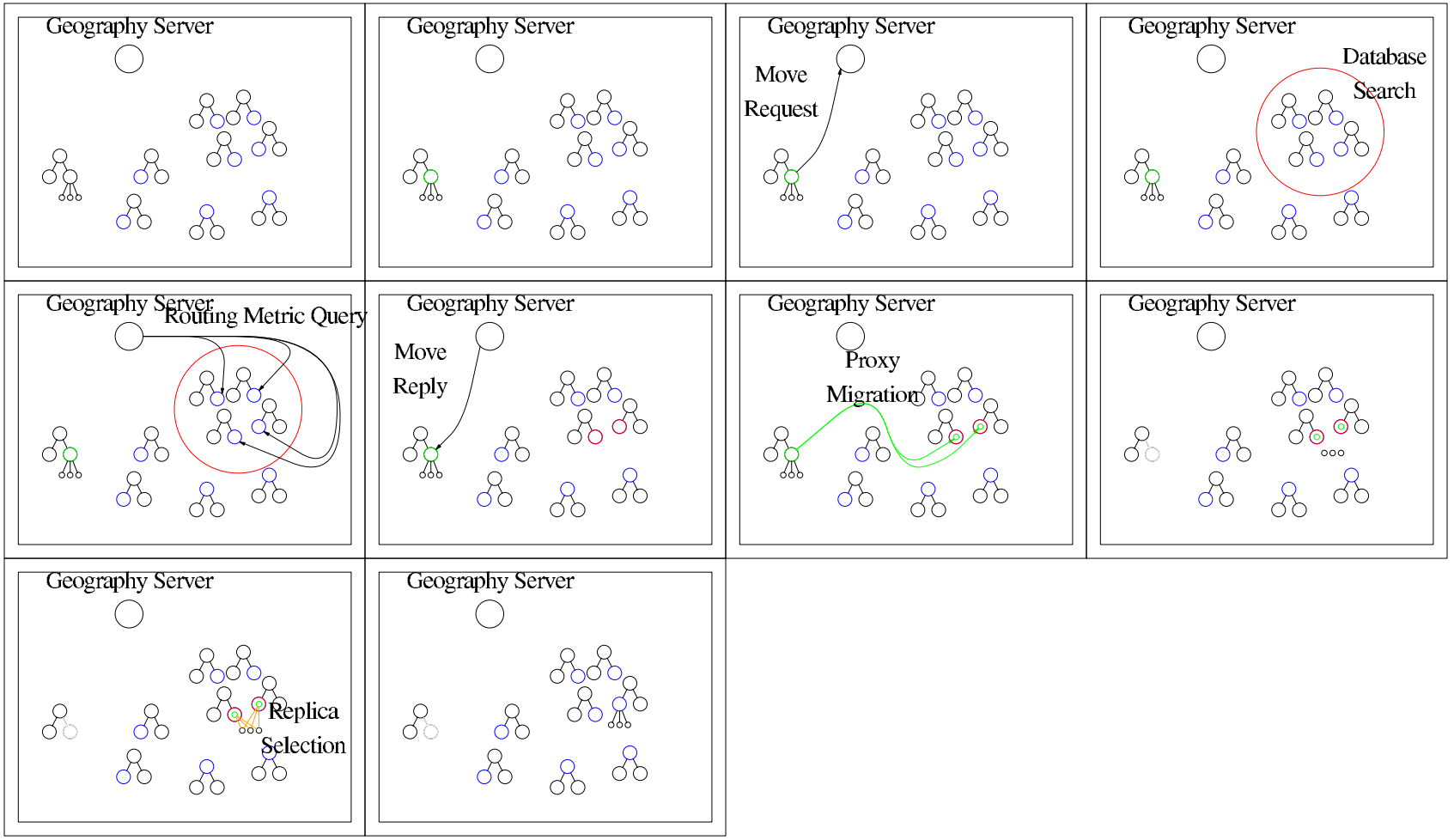
The Routing Hierarchy

- use request routing metrics
- select several candidates
- start several provisional proxies

Replica Selection

- provisional proxies look like mirrors
- arriving clients pick closest proxy

Mobility: The Movie



Proxy Encapsulation

All I need is this multicast address...

Encapsulation: By Any Other Name

- simplified process migration

Proxy State

- cache contents
- caching strategy
- multicast channels

Representation (heterogeneity)

- language-based?
- structure-based?

Wei Yue is exploring these options

Conclusions

Mobility support is key to info mgmt

LSAM supports mobility

Prototyping is underway

Initial results are good

Future Work

Where to we go from here?

Integration

- put mobility parts together
- integrate with LSAM

Trend Analysis

- "I go home every day at 5:00"
- "I have a P.I. meeting in D.C."

Geographic multicast using soft state?

This seminar brought to you entirely by ASCII-based tools