CSci 8980
Mobile Cloud Computing

Outsourcing: Components
Cloud as an Outsourcing Platform

• Benefits:
  – Scalability, elasticity: Can scale up to large number of users on-demand
  – Sharing: Can share data and computation across multiple users/applications

• Problem: Network latency
Mobile Outsourcing to Cloud: Key Questions

• **Question 1**: Is cloud a feasible mobile outsourcing platform despite higher network latencies?

• **Question 2**: How can we detect and exploit sharing across applications for better performance and efficiency?

• **Our Focus**: Cloud-side optimizations for efficiency and performance of mobile outsourcing
Motivation

- **Mobile-to-Cloud Outsourcing Platform**
- Feasibility of Cloud for Outsourcing
- Sharing-aware Application Placement
- Experimental Evaluation
- Conclusion
Mobile-to-Cloud Outsourcing Platform

- Code Repository
- Cloud (Amazon EC2)
- Offloading Servers (VMs)
- Component Manager
- Mobile Users (Android)
Feasibility of Cloud for Outsourcing

- Setup: Android HTC Hero phone, 1 small EC2 instance
- Network latency: WiFi: 82ms, 3G: 151ms, wired: 44ms
- Application: Image processing - Blur Filter

- Cloud is desirable for compute-intensive applications
Data Sharing Between Applications

• Applications often share common data
  – E.g.: Intermediate data between Image processing and Face detection
Data Sharing with Outsourcing

• Data sharing method determined by application locations
Impact of Application Location on Performance

- Desirable to co-place sharing applications together
Sharing-aware Application Placement

- Detects the application components that have potential to share data
- **Association analysis**: Sequence mining on temporal data access
- Co-locate such components on same server
Other Application Placement Strategies

• User-centric
  – Mobile users are assigned dedicated servers
  – No users share servers
  – Good for cross-application data sharing

• Application-centric
  – Components are assigned to dedicated servers
  – Multiple users can share the same component on the same server
  – Good for resource utilization
Dynamic Provisioning

• Scale the number of servers
  – According to CPU utilization state
  – High/Low thresholds for detecting overloaded/underloaded servers
• Create server: when all servers are overloaded
• Merge server: when a server is underloaded
Outline

✓ Motivation
✓ Mobile-to-Cloud Outsourcing Platforms
✓ Feasibility of Cloud for Outsourcing
✓ Sharing-aware Application Placement
  • Experimental Evaluation
  • Conclusion
Experimental Setup

- Amazon EC2 offloading platform
  - Component manager, code repo: small instances
  - Offloading servers: Micro instances
- Emulated mobile workload
  - 100 users, 10 image processing components
  - Some requests share data among components
- Metrics
  - Server utilization: measures resource efficiency on the cloud
  - Network traffic: measures user performance and outsourcing overhead
Server utilization

- Sharing-aware Co-location is close to App-centric in efficiency
Sharing-aware Co-location is close to User-centric (and optimal) in network overhead
Sharing-aware placement achieves high co-location rate
Dynamic Provisioning

- Num servers varies according to workload
Conclusion

• Cloud is a desirable outsourcing platform for mobile computation
• Cloud provides cross-application sharing opportunities as well as dynamic scalability
• Sharing-based application placement can achieve high efficiency as well as performance