Dynamically Partitioning Applications between Weak Devices and Clouds

Mobile Cloud Computing and Services Workshop 2010

Byung-Gon Chun, Petros Maniatis
Intel Labs Berkeley
Weak devices

- Weak devices
  - Smartphones
  - Netbooks/Tablets
  - Mobile Internet Devices
  - Embedded devices
  - Vehicles
  - TVs
  - ...

- Applications
  - Facebook, Twitter
  - Video, photo viewing/editing
  - Finances
  - Games
  - Rich media
  - ...

Byung-Gon Chun, Mobile Cloud Computing and Services Workshop, June 15, 2010
CloneCloud (HotOS 2009)
Augmenting mobile device execution through cloud execution
Current applications are statically partitioned
No single partitioning fits all due to heterogeneity

- Diverse environments
  - Device platform
  - Network
  - Cloud
- Diverse workloads

Dynamic Partitioning

Dynamically shifting computation between weak device and cloud
Example application for dynamic partitioning: image matching

Image feature generation

Input image
- Keypoint extraction
- Local computation
- Descriptor computation

Output images
- Classification
- Similarity calculation

SIFT (IJCV 2004), SURF (ECCV 2006)
Workload

- Size of the image
- Content of the image
- Database size

Image feature generation

Nearest neighbor search
Device platform

- CPU: 500MHz w/o FPU <-> 2GHz w FPU
- Memory: 192MB <-> 2GB
- Persistent store: 256MB <-> 160GB
- Special units: GPU
- Charging mode
- More processing power, more memory
  -> More processing at the client
Network

- 3G, WiFi
- Wimax, LTE
- Connection variation
- Faster networks -> send raw data and offload more processing to the server
Cloud

- Availability
- Distance
- Cost
Problem formulation

- Application: a set of modules interacting each other
Problem formulation

- Partitioning: multiple sets of modules that cover all application modules
Partitioning example

Weak device

Cloud

Byung-Gon Chun, Mobile Cloud Computing and Services Workshop, June 15, 2010
Partitioning example

(E.g., 1 and 2 should be co-located)
Partitioning example

Weak device

1 2

Cloud

3 4

(E.g., optimize energy consumption)
Optimization

- Correctness
- Metric optimization
Optimization

- Minimize $\sum_{i \in I} C(i)$ where

\[
C(i) = \sum_{m \in P_d} C_p(m, d, i) + \sum_{m \in P_s} C_p(m, s, i) + \sum_{m_1 \in P_d, m_2 \in P_s} C_c(m_1, m_2, t, i)
\]

subject to

\[
m = P_d, \forall m \in L_1(d)
\]

\[
m = P_s, \forall m \in L_1(s)
\]

\[
m_1, m_2 \in P_d \text{ or } m_1, m_2 \in P_s, \forall (m_1, m_2) \in L_2
\]
System support for dynamic partitioning

- Application structuring
- Partitioning choice
- Security of dynamic partitioning
Application structuring

- Both client and server have all parts of the application
- Instantiate what modules to run at client and server dynamically at run time
- Wire them flexibly
Partitioning choice

- Policy
  - Goals: execution time, energy consumption, total money spent, security

- Mechanism
  - Profile costs
  - Online partitioning decision - prediction, fast optimization
Security of dynamic partitioning

- A module containing sensitive data of a machine does not run at another machine
- Automated approaches of privacy preserving partitioning
Case study: partitioning of mobile device applications

- Offload computation from mobile devices to servers: an example of C-S program partitioning
- Recent fine-grained partitioning proposals
  » CloneCloud (HotOS 2009)
  » MAUI (Mobisys 2010)
Case study: CloneCloud v1

- Application structuring
  - Create (trusted) clones
  - Method-level partitioning
  - Offload execution of partitioned methods

- Partitioning
  - Profiling
  - Program analysis
  - ILP optimization
Case study: CloneCloud v1

- Adaptation – prediction using ML
- Applications: image search, virus scanning, privacy-preserving targeted advertising
Related work

- Static partitioning of C-S programs – Links, Hops, Hilda
- Programmer-assisted partitioning – Spectra, Chroma, Odyssey, Protium
- Automatic partitioning – Coign, Wishbone
- Class-level partitioning – AIDE, OLIE
- Partitioning via information flow types - Swift
Summary

- Present dynamic partitioning between weak device and cloud in diverse environments and workloads
- Formalize the dynamic partitioning problem
- Discuss its system support
Thank you!
Questions?