IMCM: Actor-Based Adaptive Framework for Mobile Hybrid Cloud Applications

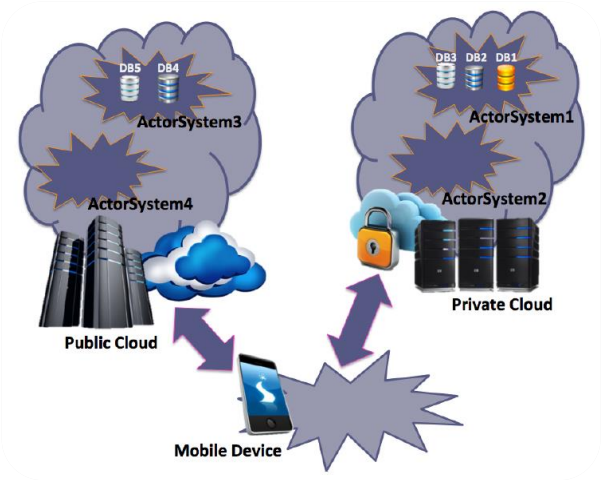
Reza Shiftehfar, Kirill Mechitov and Gul Agha

Outline

- Mobile hybrid cloud framework
- Lightweight system monitor
- Policy-based distribution
- ACC demo scenario

Mobile cloud computing (MCC)

- Mobile applications are growing increasingly complex and resource-heavy
- Want to make use of cloud resources to overcome limitations of mobile devices
  - Hardware
  - Network access
  - Energy
- How? Use code offloading



MCC issues

How to implement code offloading?

- Rewrite application code
  - For specific cloud provider?
- Transparent full VM emulation in the cloud
  - High overhead, data use
- Some combination
  - Fine-grained offloading
  - Separation of concerns

How to deal with multiple cloud providers, including private clouds?

- Code migration and data access restrictions
- Security and privacy considerations

Related work

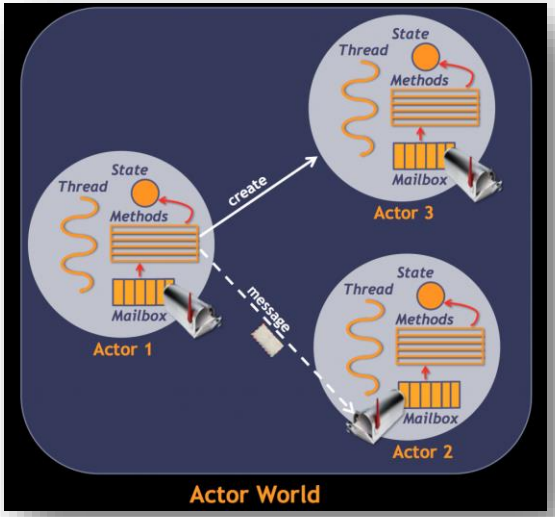
Year	System Name	Goal	Offloading Decision	Partition Level	Parallel	Policy-based Security/ Privacy	Manual Work	No. Cloud spaces
2010	MAUI	Mobile Energy Saving	Dynamic	Method	No	No	Yes	1
2011	CloneCloud	Mobile Energy Saving = Performance Improvement	Static	Method	Pseudo	No	No	1
2012	ThinkAir	Mobile Energy Saving = Performance Improvement	Dynamic	Method	Pseudo	No	Yes	1
2012	Cloud OS (COS)	Load Balancing for Cloud space	Dynamic	Actor	Yes	No	No	Can be Many
2015	IMCM	Mobile Energy Saving, Performance Improvement, Combination for Applications	Dynamic	Actor	Yes	Yes	No	Many

IMCM: mobile hybrid cloud framework

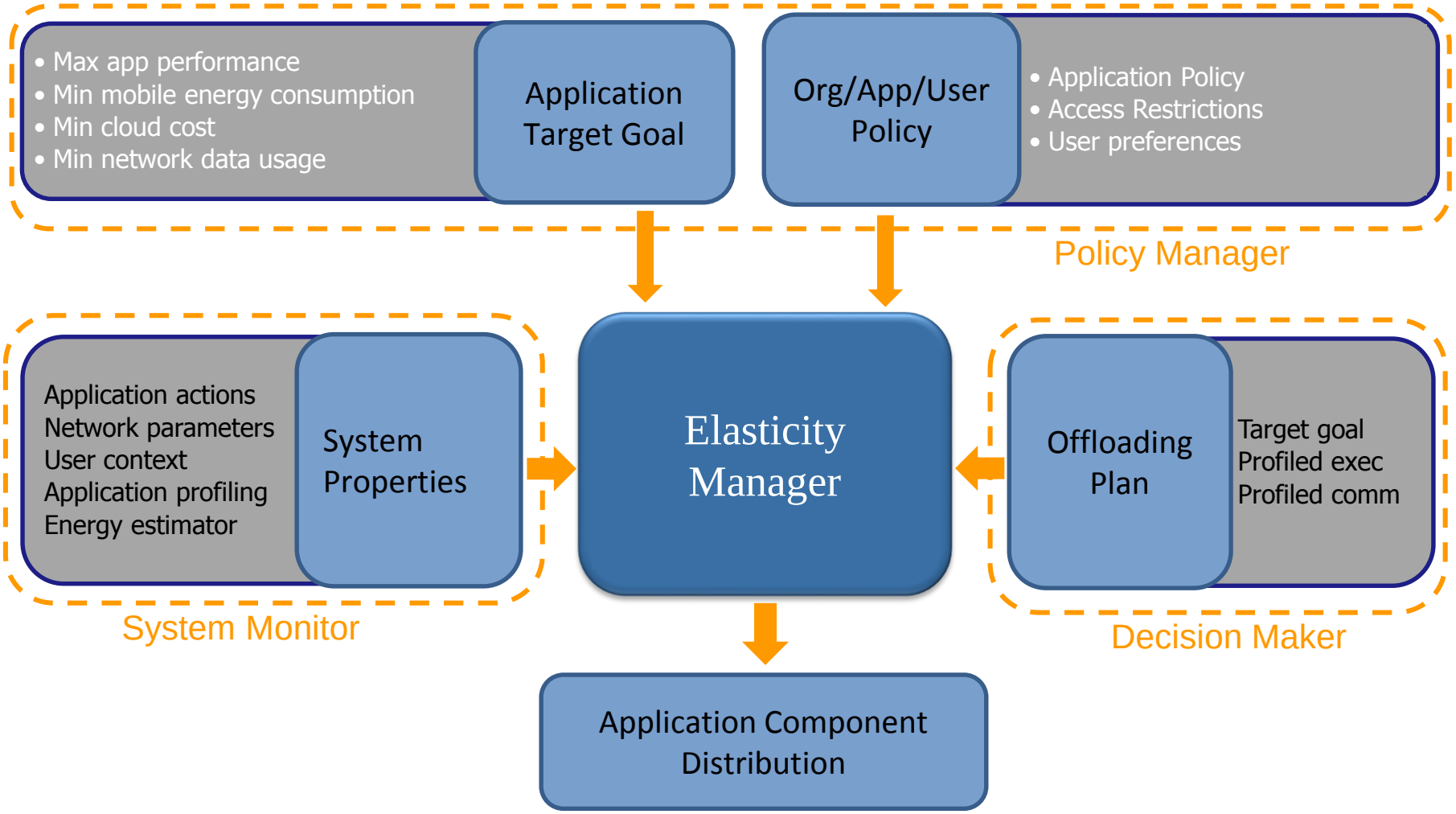
- Bridge the gap between mobile application and cloud computing
- Separate application logic from component distribution
- Allow dynamic distribution of components based on run-time changes
- Support policy-based definition of customized constraints

IMCM design and implementation

- Actor programming model
  - Natural Concurrency
  - Decentralization
  - No data races
  - Elasticity
  - Ease of scaling-up or -out
  - Location transparency
    -> Transparent migration
- Implementation: SALSA
  - Full actor semantics
  - Lightweight actors
  - Migration support
  - Java-based (portable)

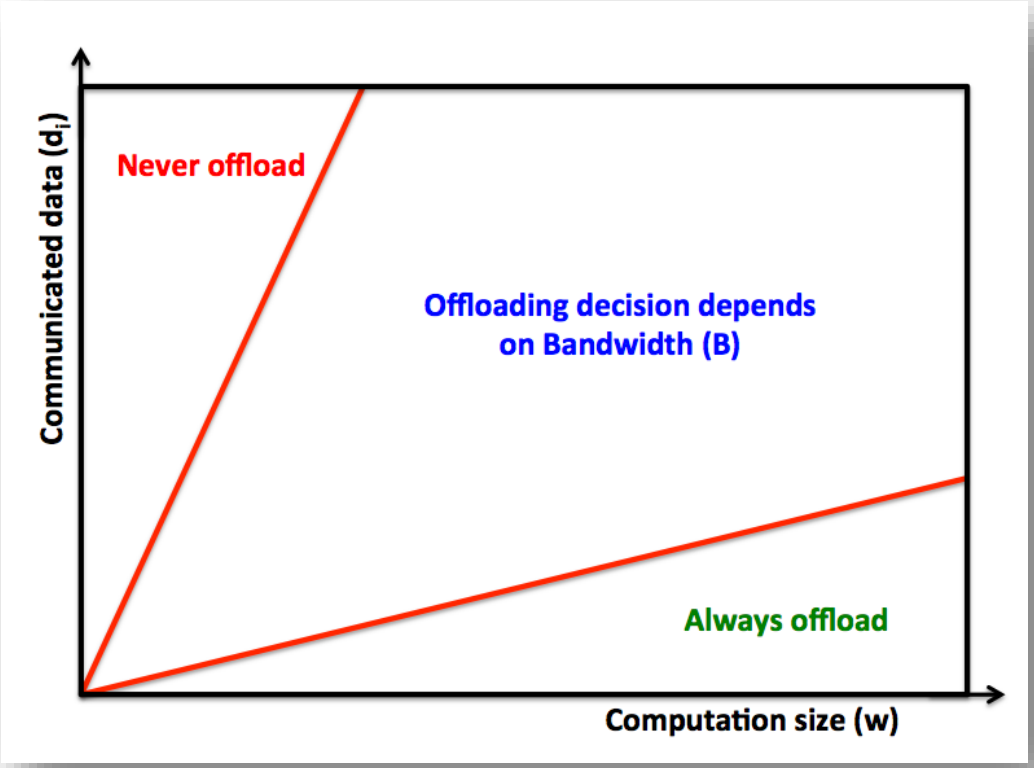


IMCM system overview

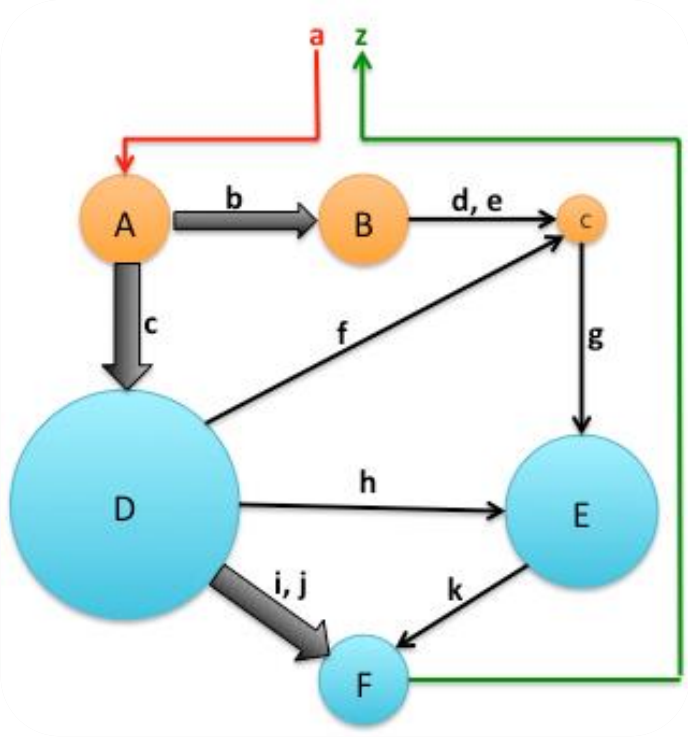


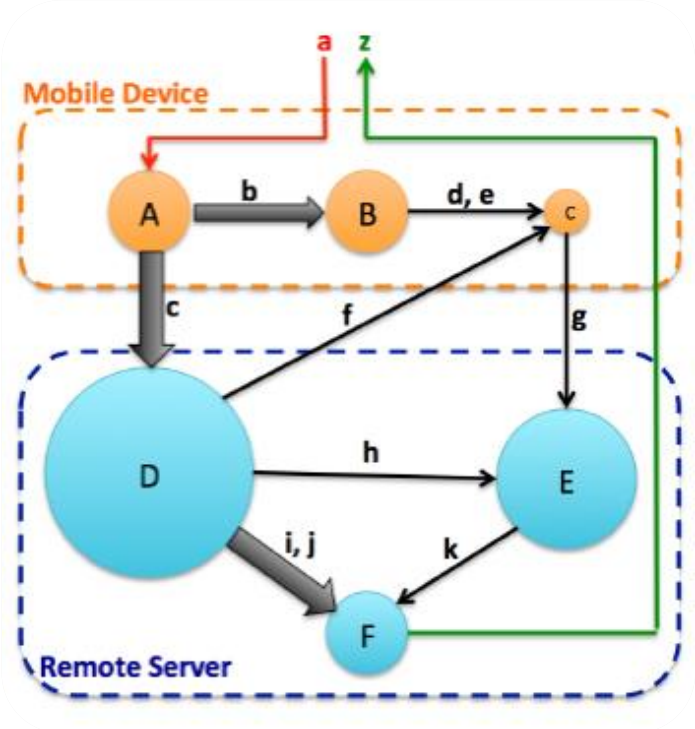
Offloading decision: bandwidth

$$\frac{w}{S_m} > \frac{d_i}{B} + \frac{w}{S_s} \quad \Rightarrow \quad w * \left(\frac{1}{S_m} - \frac{1}{S_s}\right) > \frac{d_i}{B}$$



Offloading decision: parallelism





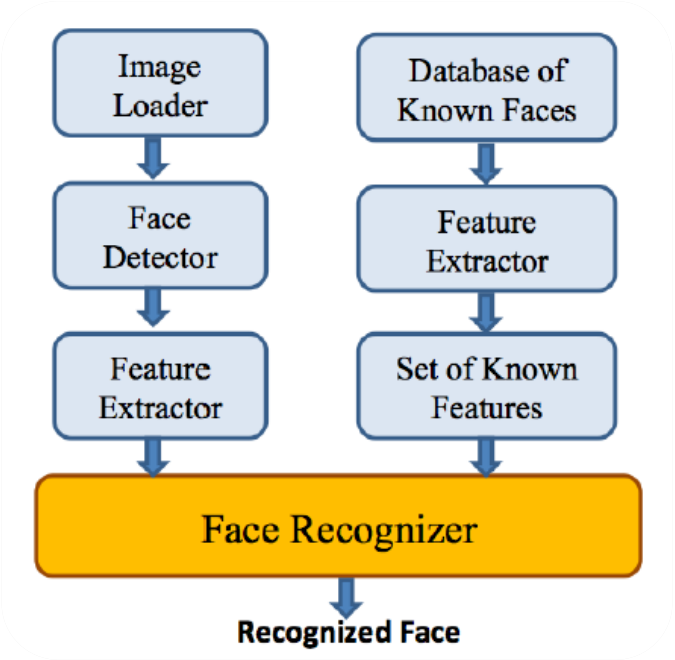
Some benchmark results

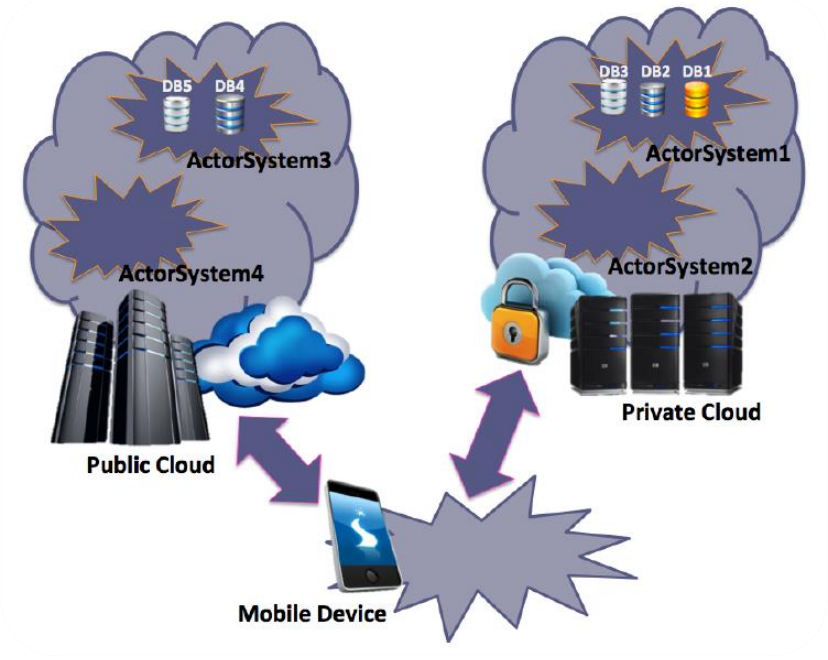
- Running the same code on a faster machine
- Running some components in parallel
- Keeping in mind the cost of offloading

Experiment	Application Characteristic				Raw Speedup	Offload Speedup
	Comp.	Comm.	I/O			
			read	write		
NQueen	intensive	-	-	-	73	56
Image	intensive	limited	limited	-	91	44
Trap	intensive	limited	-	-	30	21
Virus	-	-	intensive	-	28	21
Rotate	-	-	intensive	intensive	28	9
ExSort	intensive	-	intensive	intensive	46	36
Heat1	limited	medium	-	-	31	29
Heat2	limited	high	-	-	14	14

Example: face recognition app

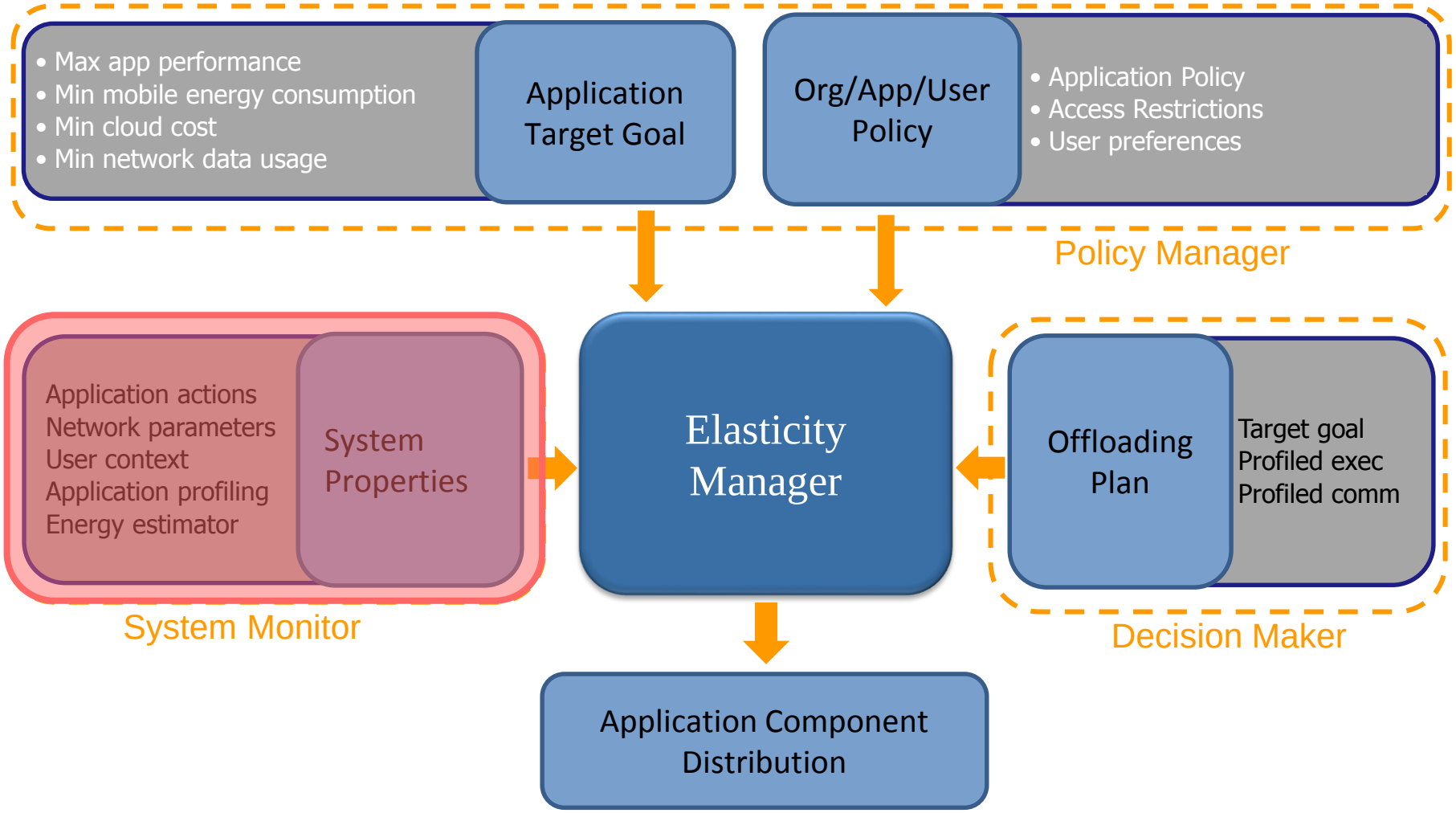
Example: face recognition app





- Components with different computation, bandwidth, energy characteristics
- Some data inputs/outputs may be restricted to specific systems or users

IMCM system overview



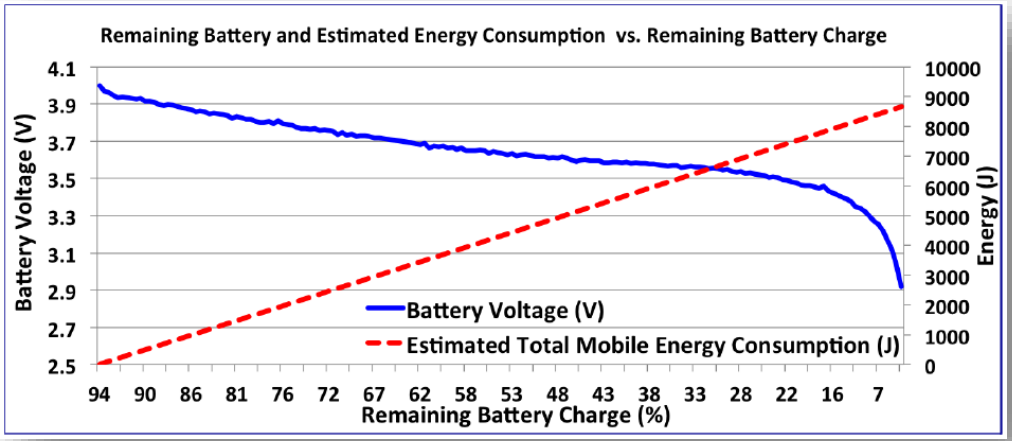
IMCM system monitor

- What to monitor?
  - Application actions
    - At the level of actor primitives (create, migrate, send/receive)
  - System actions
    - Offloading decisions, costs
  - System/environment conditions
    - Performance, energy, connection speed, resource availability
- When to notify?
  - Changes in monitored parameters
  - Changes in org/user/system policies
  - Violation of existing policies

Example: face recognition app

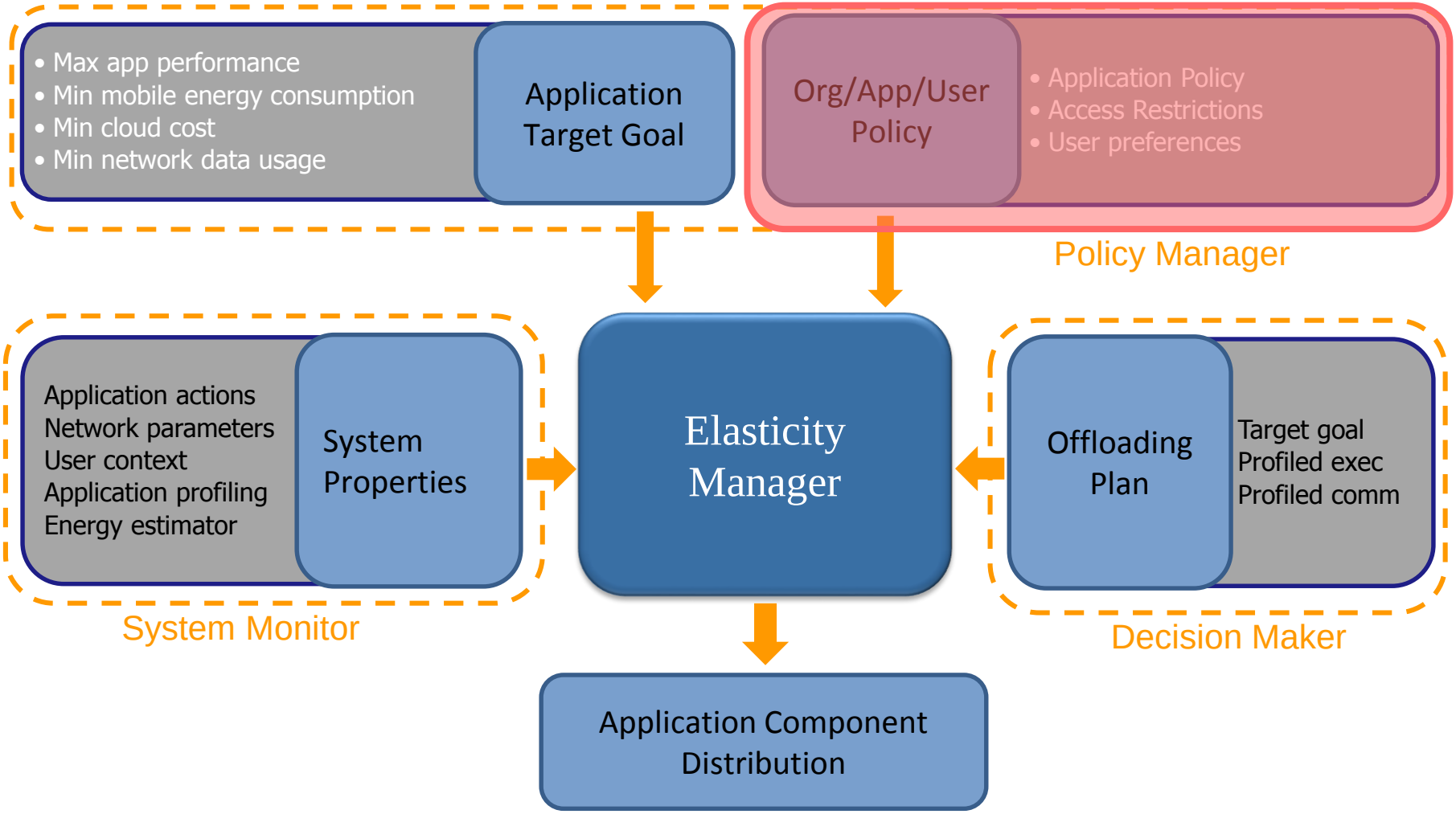
- Energy use estimation based on simple model

Time (sec)	Voltage Drop (V)	Energy (mJ)	No. Face Detector	No. Feature Extractor
40	0.0324	58350	10	10
100	0.0313	56340	7	10
161	0.0187	33700	3	8



Component	Energy Consumption (mJ)
Face Detector	2668
Feature extractor	2019

IMCM system overview



Flexible policy-based restrictions

- Complete security solution requires:
  - Authentication
  - Authorization (access control)
  - Auditing

- Privacy issues
  - Entities involved
    - Owner organization
    - Programmers/developers
    - End-user
  - NIST SP 800-144: Guidelines of Security & Privacy of Public Cloud

IMCM rule definition & enforcement

- Requirements
  - Fine granularity
  - Rules defined by organization, developer, end-user
  - Covers all possible actions

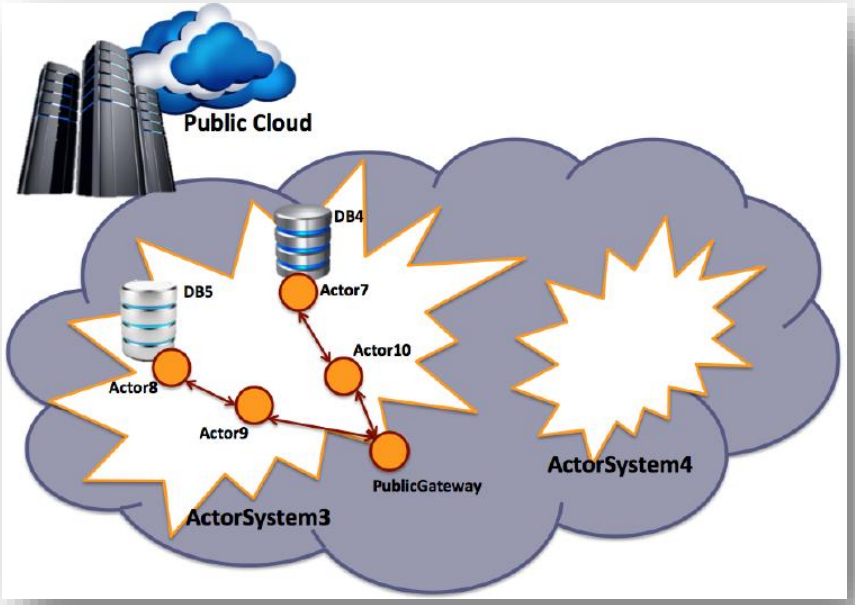
- Action control system based on:
  - Attributes of the requester
  - Attributes of the resource
  - The requested action
  - The specified policy

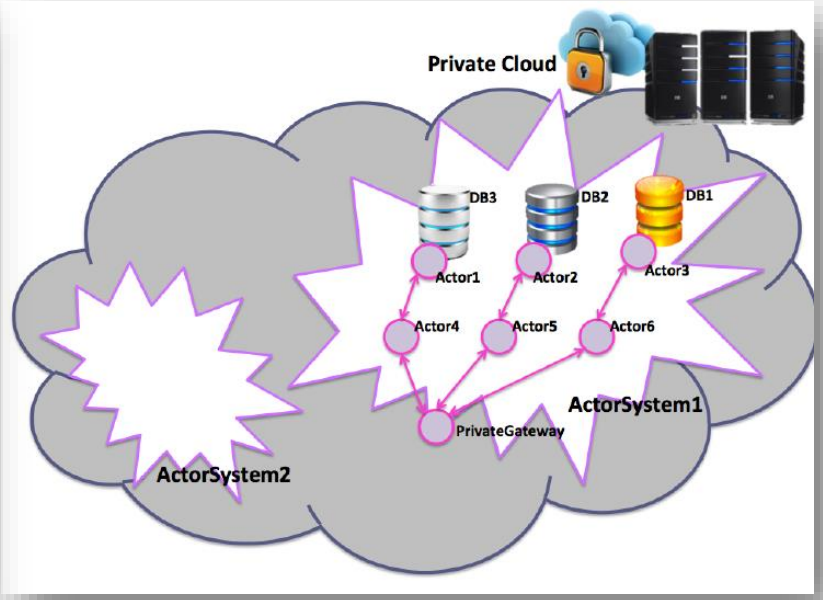
IMCM rule definition & enforcement

- Authorization policy
  - Hard
    - Organization-wide policy
  - Soft
    - User/developer policy (app-instance-specific policy)
- Every application-instance comes with:
  - An immutable hard policy
  - A customizable soft policy
- Grammar to define rules
  - Attribute based
  - Static & dynamic binding

Example: face recognition app

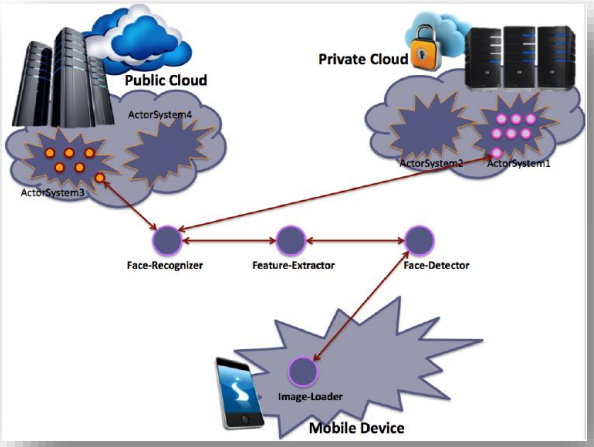
- Hard policy
  - Different users:
    - Security personnel
    - General public



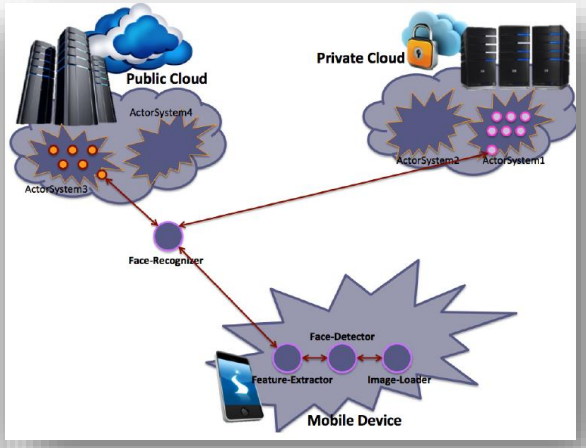


Example: face recognition app

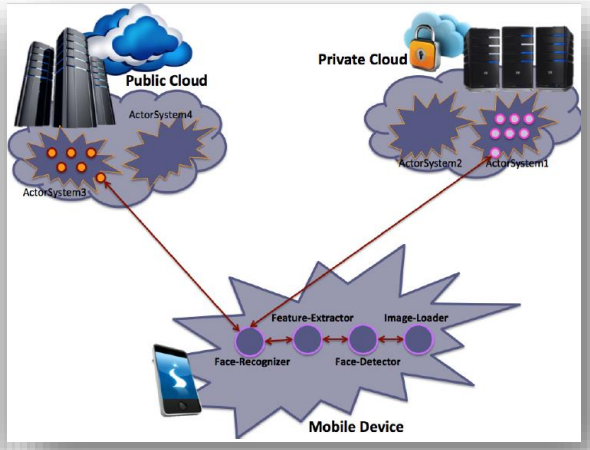
- Soft policy
  - Different user expectations:
    - Type 1: No additional privacy concerns
    - Type 2: Privacy concerned users
    - Type 3: Extremely cautious users



Type1



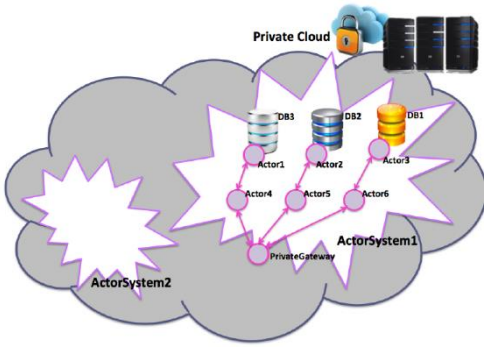
Type2



Type3

Example: face recognition app

- Sample private cloud rules for restricted code and data access
  - E.g., no send, migration from private to public cloud



1. ActorSystem: {Name:ActorSysPrivate1, Static (URL:174.123.78.456, Port:1362))}
2. Actor: {Name:ActorPrivateGate, Static (Reference: akka.tcp://app@174.123.78.456/privateGateway, ActorSystem:"ActorSysPrivate1)}
3. Actor: {Name:ActorPrivateVisasDB, Static (Reference: akka.tcp://app@174.123.78.456/Actor1, ActorSystem:ActorSysPrivate1)}
4. Actor: {Name:ActorPrivateResidentDB, Static (Reference: akka.tcp://app@174.123.78.456/Actor2, ActorSystem:ActorSysPrivate1)}
5. Actor: {Name:ActorPrivateCitizenDB, Static (Reference: akka.tcp://app@174.123.78.456/Actor3, ActorSystem:ActorSysPrivate1)}
6. Actor: {Name:ActorPrivateVisaProcessor, Static (Reference: akka.tcp://app@174.123.78.456/Actor4, ActorSystem:ActorSysPrivate1)}
7. Actor: {Name:ActorPrivateResidentProcessor, Static (Reference: akka.tcp://app@174.123.78.456/Actor5, ActorSystem:ActorSysPrivate1)}
8. Actor: {Name:ActorPrivateCitizenProcessor, Static (Reference: akka.tcp://app@174.123.78.456/Actor6, ActorSystem:ActorSysPrivate1)}
9. AnonymousActors: {Name:AnonymousPrivate1, Ref-ActorSystem:ActorSysPrivate1, Existence:FORBIDDEN}
10. AnonymousActorSystems: {Name: Other-ActorSys-Private, URL:174.123.78.456, Creation: FORBIDDEN}
11. Rule: {Name:Private-Rule-1, Subject (ActorSystems:ActorSysPrivate1), Object (ALL), Actions: ALL, Permission: DISALLOWED}
12. Rule: {Name:Private-Rule-2, Subject (ActorSystems:ActorSysPrivate1), Object (ActorSystems:ActorSysPrivate1), Actions: SEND-TO, RECEIVE-FROM, Permission: ALLOWED}
13. Rule: {Name:Private-Rule-3, Subject (Actor:ActorPrivateGate), Object (ALL), Actions: SEND-TO, RECEIVE-FROM, Permission: ALLOWED}
14. Rule-Order: {Name: Private-Rule-Order-1, Subject (Rules: Private-Rule-2), Object(Rules: Private-Rule-1), Order: PRECEDENCE}
15. Rule-Order: {Name: Private-Rule-Order-2, Subject (Rules: Private-Rule-3), Object(Rules: Private-Rule-1), Order: PRECEDENCE}
16. ActorSystem: {Name:ActorSysPrivate2, Static (URL:174.123.78.456, Port:1369))}
17. AnonymousActors: {Name:AnonymousPrivate2, Ref-ActorSystem:ActorSysPrivate2, Existence:ALLOWED}
18. Rule: {Name:Private-Rule-4, Subject (ActorSystems:ActorSysPrivate2), Object (ALL), Actions: ALL, Permission: ALLOWED}

IMCM: Summary

- Provides an adaptive solution for mobile cloud application development
- Allows programmers to focus on application logic
- Requires minimal manual tuning
- Provides full parallelism
- Supports different target offloading goals
- Allows use of multiple hybrid cloud spaces
- Allows customizable privacy policies for users
- Allows mandatory security policies for organizations

ACC demo scenario

ACC demo scenario

- Often the first information from a disaster area comes from mobile photos and videos

- Some of this stream of data may be of use to first responders
  - Damage assessment
  - Location of downed power lines
  - Location of people to be evacuated
  - Etc.

ACC demo scenario

- Users can run an app that will automatically share some relevant information with the responders
  - User concerns:
    - Privacy
    - Energy use
  - Other concerns:
    - Bandwidth
    - Relevance
- Military may make some private cloud resources available for this to supplement/replace public clouds
  - Security concerns:
    - Data leaks
    - Attacks