

DynaMix: dynamic mobile device integration for efficient cross-device resource sharing

Chae, Dongju, et al.
"DynaMix: dynamic mobile device integration
for efficient cross-device resource sharing. "
2018 {USENIX} Annual Technical Conference
({USENIX}{ATC} 18)



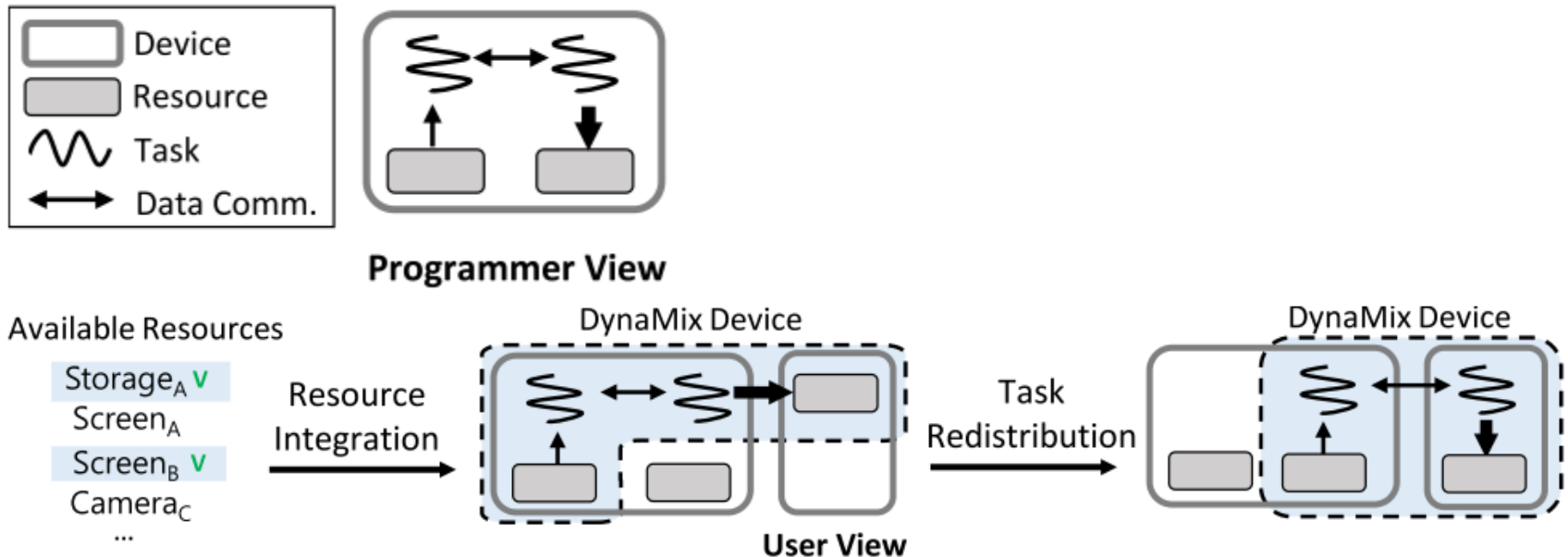
Introduction

- IoT 장치가 널리 사용되면서 사람들은 수많은 서로 다른 장치를 사용
- 이러한 환경은 다양한 리소스를 동시에 활용하는 멀티 디바이스 서비스에 대한 요구를 만듦

Introduction

- 현재 존재하는 크로스 디바이스 리소스 공유 방식은 여러 문제가 있음
 - 효율적인 파티셔닝에 대한 짐을 개발자에게 부과함
 - 모든 연산 리소스와 I/O 리소스의 공유를 동시에 지원하지 않음
- 논문은 DynaMix라는 효율적인 크로스 디바이스 리소스 공유를 위한 동적 모바일 장치 통합 프레임워크를 제안
 - 새로운 플랫폼
 - 높은 응용성
 - 높은 성능
 - 높은 신뢰성

DynaMix Framework



- DynaMix는 특별한 프로그래밍 개념이나 API를 요구하지 않음
- 응용이 잘 동작할 수 있도록, 실행 중에 적절한 리소스를 선택
- 응용 작업들은 자유롭게 원격 자원들을 접근,
최적의 작업 재분배를 위해 마이그레이션 될 수 있음

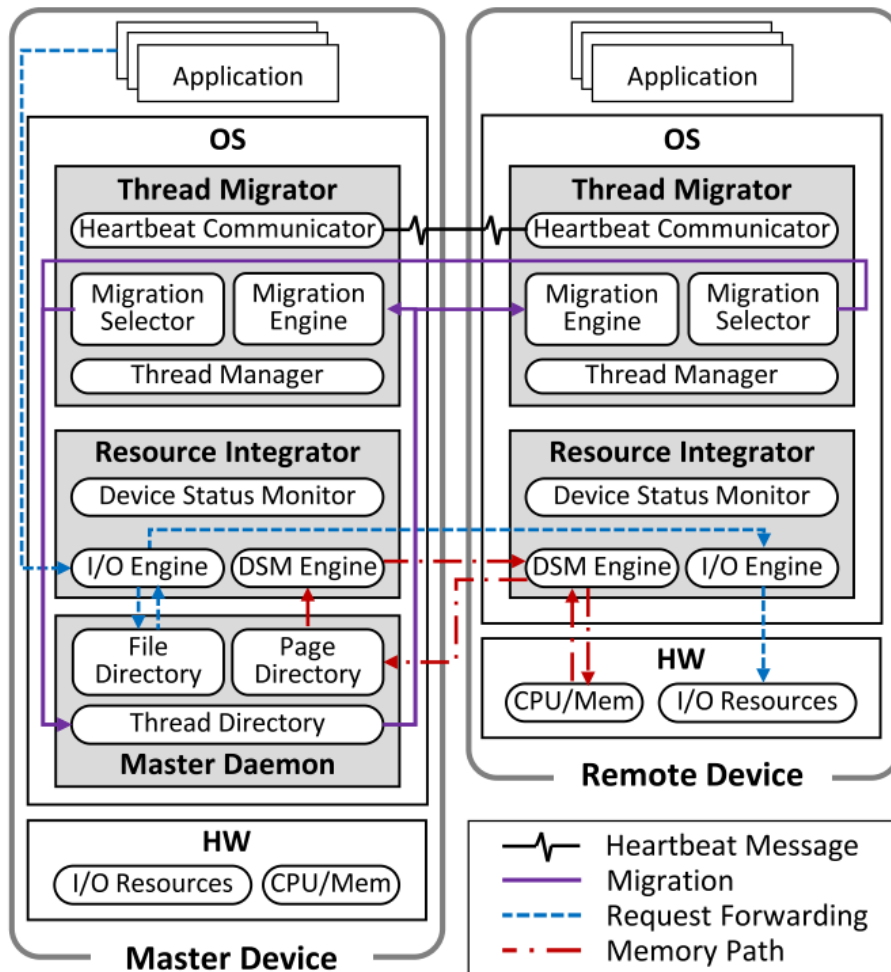
Remote Resource Integration

- 연산 리소스와 I/O 리소스를 통합 관리해 단일 장치에서 프로그래밍하는 것과 같은 효과를 목표로 함
 - 원격 리소스에 대한 정보를 모으고, 그 리소스들을 응용 프로그램이 활용할 수 있도록 함
 - 응용 프로그램이 원격 자원을 사용하려고 한다면, resource integrator는 대상 resource integrator에 요청을 전송
 - 대상 resource integrator는 연산 결과, 혹은 I/O 실행 결과를 응용 프로그램에 전송

Resource-aware Task Redistribution

- DynaMix는 장치간 트래픽을 최소화하기 위해 스레드를 최적의 장치로 이동시키는 Resource-aware task redistribution을 적용
 - Resource integrator가 리소스의 경쟁을 탐지
 - 경쟁이 감지되면 thread migrator는 스레드 재분배 시나리오 간의 tradeoff를 비교하고 최적의 시나리오를 찾음
 - Thread migrator는 최적의 시나리오에 따라 스레드를 마이그레이션 함

Implementation



■ Resource Integrator

- DSM Engine
- I/O Engine
- Device Status Monitor

■ Thread Migrator

- Thread Manager
- Migration Selector
- Migration Engine
- Heartbeat Communicator

■ Master Daemon

- Thread Directory
- Page Directory
- File Directory

Resource Integrator

▪ DSM Engine

- 여러 장치의 메모리 영역을 Distributed Shared Memory 방식으로 단일 메모리처럼 관리
- 성능 최적화를 위해 다음 방법을 채택
 - Lazy Release Consistency
 - 순차 패턴을 발견하면 memory prefetch
 - Coherence overhead를 줄이기 위해 page-level coherence block을 유지

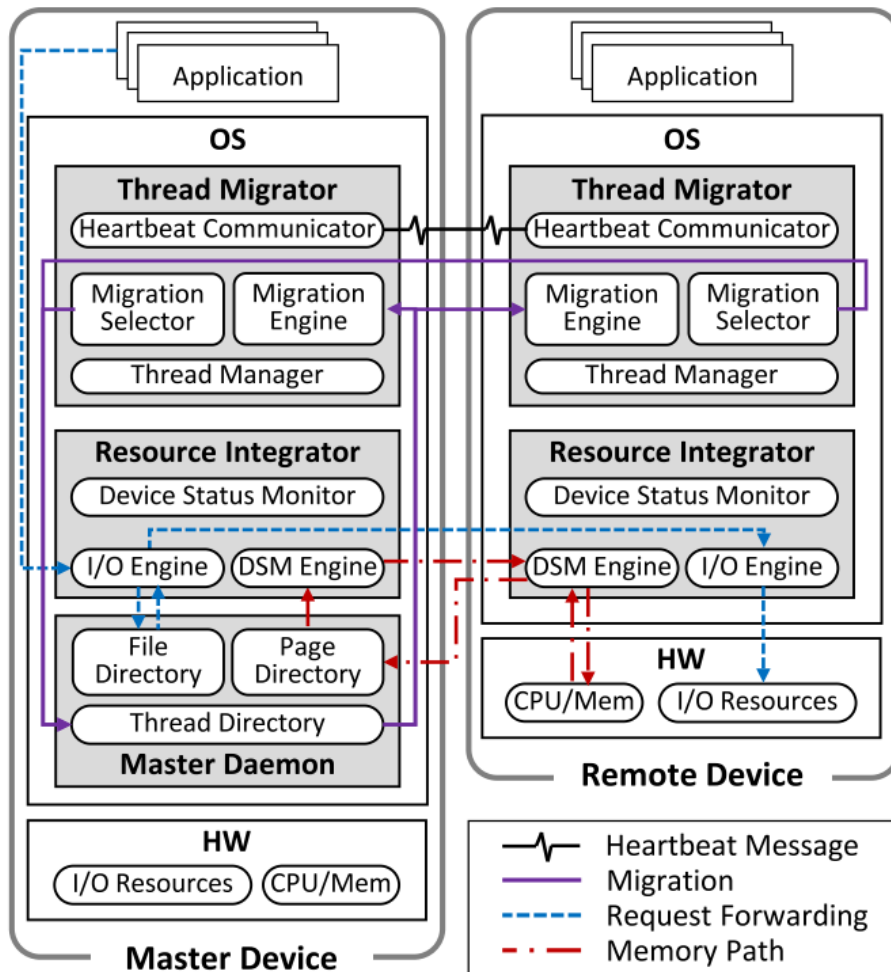
▪ I/O Engine

- I/O Engine은 로컬과 원격 I/O 리소스에 대한 접근을 커널 레벨에서 관리

▪ Device Status Monitor

- CPU와 네트워크에서의 경쟁을 탐지하기 위해 지속적으로 시스템 정보를 수집
 - 각 스레드의 CPU 사용량, 네트워크 stall time 등
- 경쟁 상황을 탐지하면 스레드 재분배가 이루어질 수 있도록 master daemon에 정보를 전송

Implementation



■ Resource Integrator

- DSM Engine
- I/O Engine
- Device Status Monitor

■ Thread Migrator

- Thread Manager
- Migration Selector
- Migration Engine
- Heartbeat Communicator

■ Master Daemon

- Thread Directory
- Page Directory
- File Directory

Thread Migrator

▪ Thread Manager

- 실행 상태, 리소스 사용량, 락과 같은 실행중인 스레드의 정보를 보유
- 리소스가 경쟁상태일 때, Thread Manager는 스레드의 데이터 통신량을 계산하고 결과를 Migration Selector로 전송
- Migration Selector는 이 정보를 기반으로 마이그레이션 될 스레드와 최적의 목적지를 결정
- 마스터 thread manager는 마이그레이션 된 스레드의 실행 정보를 체크포인트로 유지
- 의도치 않게 연결이 중단된 장치에 대한 잃어버린 스레드를 복구할 수 있음

▪ Migration Selector

- Thread manager가 전송한 정보를 가지고 마이그레이션 할 최선의 스레드와 목적지를 결정
 - 유사한 동작이 근 시일 내에 다시 일어날 것이라는 가정하에 응용 프로그램의 최근 액세스 패턴에 기반해 평가
- 네트워크 오버헤드가 최대로 감소할 수 있는 시나리오를 찾은 후 master daemon에 migration recommendation을 전달

Thread Migrator

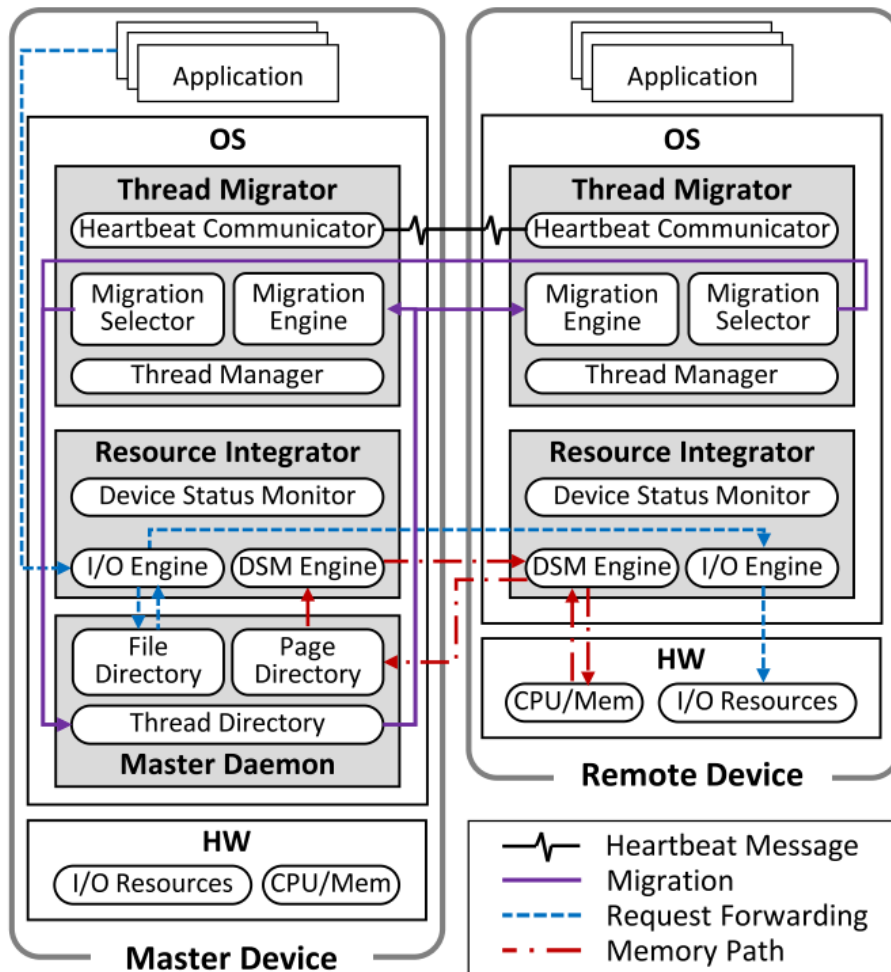
▪ Migration Engine

- Migration selector가 스레드와 목적지를 결정하면 Migration engine은 스레드 마이그레이션을 수행
- DynaMix는 스레드 복제와 라이브 마이그레이션을 통해 오버헤드를 낮춤
 - 원본 장치는 대상 스레드의 메모리 레이아웃만 대상 장치로 보내고, 대상 장치는 복제 스레드를 만듦
 - 대상 스레드가 원본 장치에서 돌아가고 있는 동안 migration engine은 대상 장치로 가장 최근에 액세스한 페이지를 전송
 - 메모리 전송이 끝나면, 업데이트 된 페이지를 갖는 대상 스레드의 실행 context를 전송(프로세스 제어 블록 등)
 - 마이그레이션이 완료되면 원본 장치의 스레드를 종료하고 대상 장치에서 스레드를 실행

▪ Heartbeat Communicator

- 주기적으로 heartbeat 메시지를 교환하여 장치 연결을 확인하고 리소스 상태를 공유
- 어떤 장치가 DynaMix에 들어오고 나가는지 탐지
- 연결이 불안정한 장치를 발견하면 master daemon에 알려 thread migration을 진행

Implementation



■ Resource Integrator

- DSM Engine
- I/O Engine
- Device Status Monitor

■ Thread Migrator

- Thread Manager
- Migration Selector
- Migration Engine
- Heartbeat Communicator

■ Master Daemon

- Thread Directory
- Page Directory
- File Directory

Master Daemon

▪ Thread Directory

- 스레드 위치와 같은 스레드의 전역 상태를 관리
- Device status monitor로부터 리소스 경쟁 신호를, migration selector로부터 migration recommendation을 수신
- 가장 높은 네트워크 이익을 얻기 위해 최적의 마이그레이션 대상을 선택한 후, migration engine을 통해 마이그레이션을 수행
- lock 정보 또한 관리해 장치끼리 교착상태에 빠지는 것을 방지

▪ Page Directory

- 메모리 동기화 작업을 조정하기 위해 메모리 페이지 공유 상태를 관리

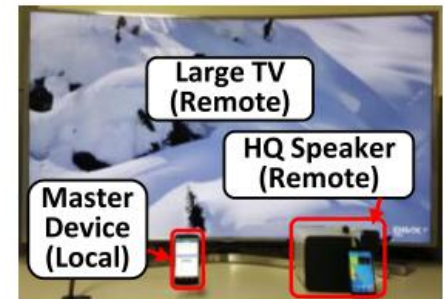
▪ File Directory

- 전역적으로 일관된 파일 관리를 위해
파일 메타데이터와 공유 파일의 물리적 위치를 관리

Evaluation

■ Home Theater

- 삼성 스마트 TV(스크린), 넥서스4(스피커), 넥서스5(마스터)
- 저장된 영상 파일을 FFmpeg을 통해 decode해서 출력



(a) Home Theater

■ Smart Surveillance

- 넥서스4(카메라), 넥서스5(마스터)
- 각 카메라로 촬영하고 있는 영상을 무작위로 하나 선정, 마스터 장치에서 Canny edge detection 알고리즘 적용



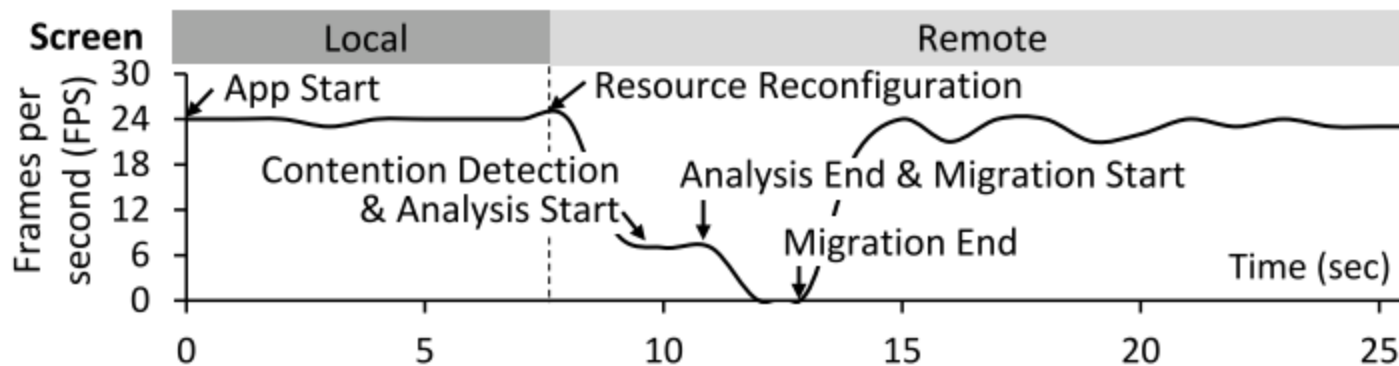
(b) Smart Surveillance

■ Photo Classification

- 넥서스 5 (최대 4대)
- 각 단말기에 4KB부터 10MB 범위의 다른 크기의 이미지를 100MB만큼 저장
- SSD_MobileNet을 이용해 이미지 분류

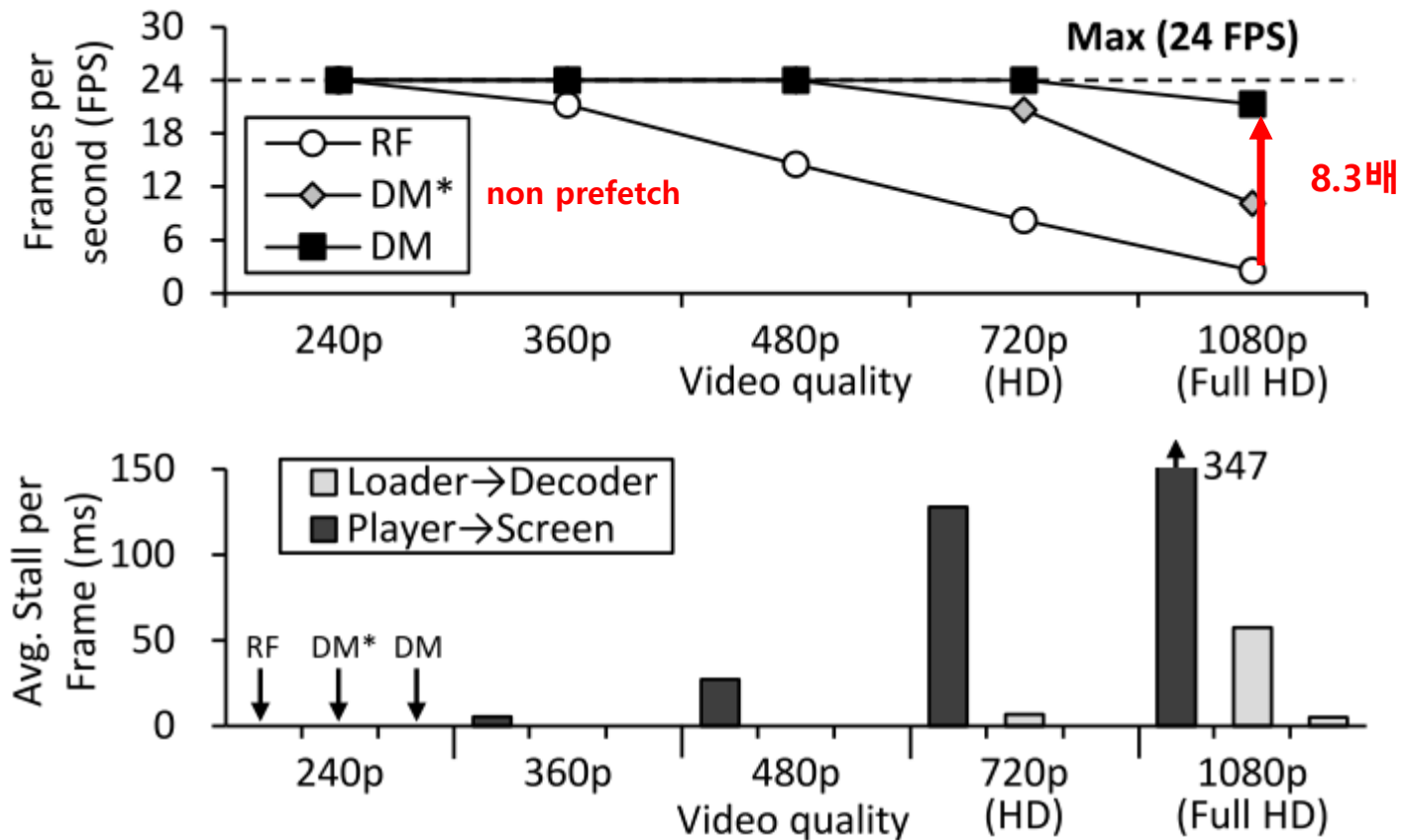
Operation Models (Home Theater)

- 720p 영상을 사용
- 처음 마스터 장치에서 영상이 출력될 때는 24fps 유지
- 원격 스크린과 원격 스피커를 인식한 후, 프레임 저하
- DynaMix를 통해 작업 재분배가 이루어지자 24fps 복귀



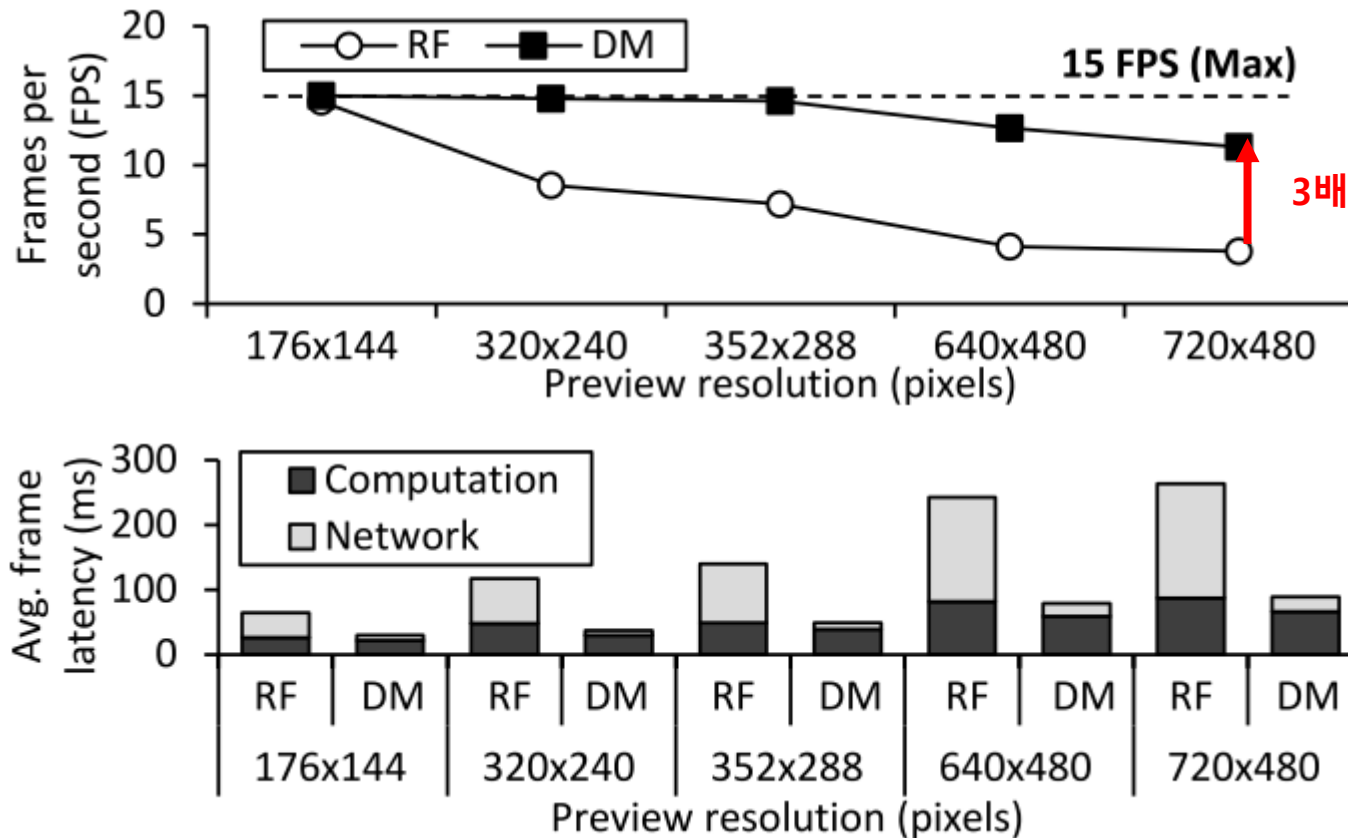
Service Quality

- Home Theater



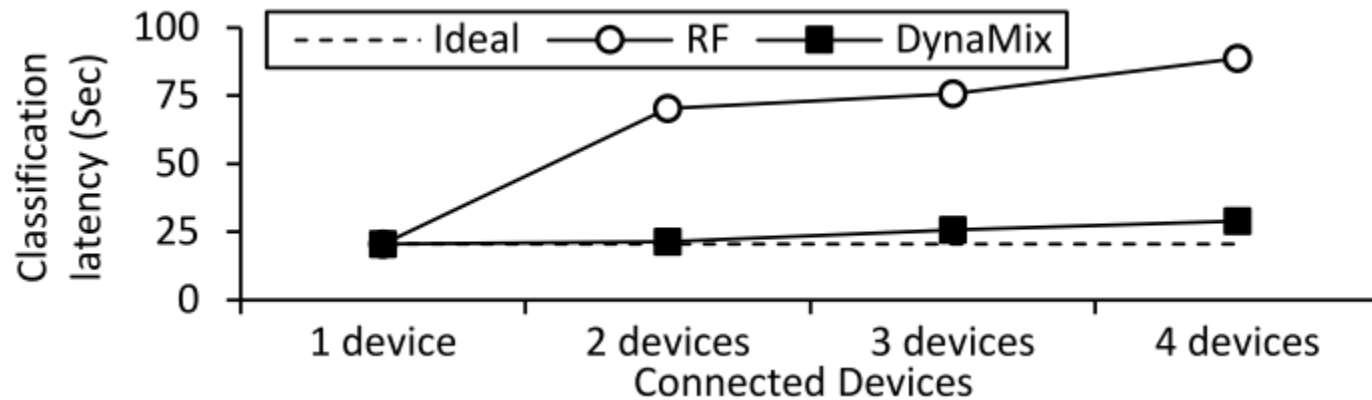
Service Quality

- Smart Surveillance



Service Quality

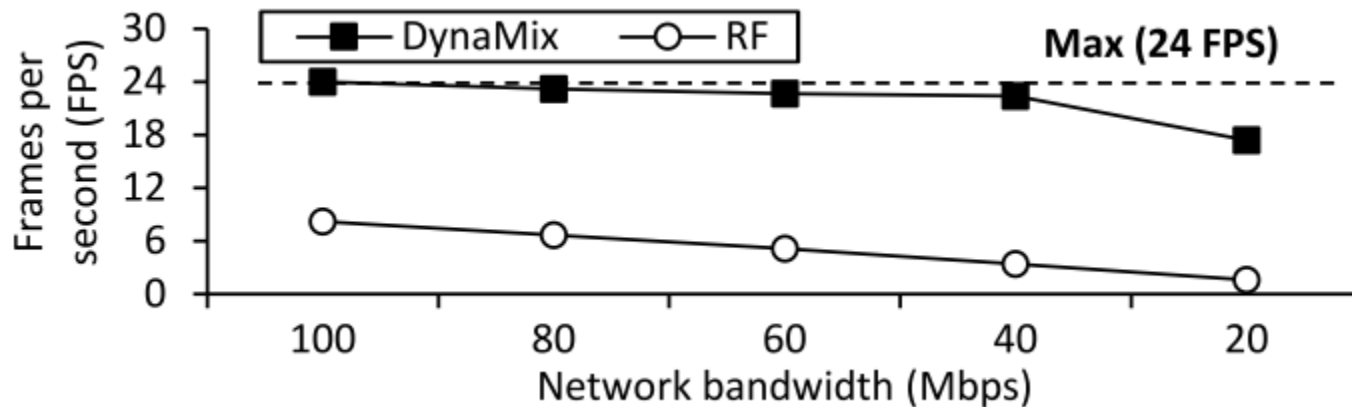
- Photo Classification



	Remote File Size (B)				
	<10K	<100K	<1M	<5M	≤10M
RF (%)	87.8	64.4	33.3	13.9	0
Mig. (%)	12.2	35.6	66.7	86.1	100

Network Sensitivity (Home Theater)

- 720p 영상을 사용



Power Consumption (Home Theater)

- DynaMix를 사용했을 때
전력 소모가 증가한 장치도 있지만
- 전체 전력 소모량은 감소함

	Average Power (mW)		
	Master Device	Screen Device	Total
RF	4985.90	5151.37	10137.27
DynaMix	2956.55	6480.51	9427.06

Conclusion

- DynaMix는 효율적인 cross-device resource sharing을 제공
- 효율적인 작업 재분배는 네트워크 트래픽을 감소시켜
멀티 디바이스 지원 응용 프로그램의 성능을 향상시킴