

실증적인 미래인터넷 연구를 위한 미디어 중심 서비스 지향 테스트베드

한상우 | 김남곤 | 차병래 | 김종원
광주과학기술원

요약

미래인터넷 인프라에 지원될 다양한 미디어 서비스들을 실험하기 위한 노력으로, 본 고에서는 미디어 중심 서비스 지향 테스트베드에 대해 소개한다. 제안하는 FIRST@PC (Future Internet Research for Sustainable Testbed @PC) 테스트베드는 PC를 기반으로 하는 저비용 시험 환경 구축과 가상화된 컴퓨팅/네트워킹 자원들을 이용한 미디어 서비스 합성 검증을 목표로 한다. 실험 방법을 구체화하기 위해, 컴퓨팅과 네트워킹 서비스들 간에 상호작용 수준에 따라 다양한 실험 프로파일과 레벨들을 제시하고, 이를 실현하기 위한 테스트베드 플랫폼 구조를 제시한다. 최종적으로 구축된 시범 테스트베드 상에서 HD 미디어 분배 실험들을 통해, 본 테스트베드의 실행 가능성을 기본적인 수준에서 검증한다.

1. 서론

태블릿 PC, 스마트폰 등 소비형 미디어 단말들이 보급되면서, 인터넷 콘텐츠들을 개인화된 형태로 이용하려는 수요가 점차 늘고 있다. 하지만 인터넷에서는 다양한 미디어 요구에 대해 유연하게 대처하기 어려운 실정인데, 이는 미디어 획득, 가공, 전송, 표현을 수행해야 하는 멀티미디어 시스템의 구조적 경직성에서 기인한다고 볼 수 있다. 이러한 문제를 완화하기 위해, 서비스 지향 구조를 수용하는 멀티미디어 시스템 구성에 관한 연구들이 진행되고 있다 [1]. 이 연구

들은 고정된 미디어 전송 시스템 구조에서 벗어나, 미디어 서비스들을 상황에 맞게 선택하여 합성하는 방식을 채택하고 있다. 먼저 사용자 요청에 대응하는 서비스 합성 틀을 준비하고, 자원과 서비스들을 적절히 대응시킨 후, 서비스들을 연결하고 동작시킨다. 이러한 방법은 웹 서비스 합성의 기본 원리와 유사하지만, 시간과 자원의 제약 하에서 데이터 전달의 연속성을 보장하기 위해 네트워킹 부분을 보다 주의 깊게 다루어야 한다. 따라서 네트워크 인프라는 연속적이고 신뢰성 있는 서비스를 제공할 수 있도록 품질 보장형 네트워킹을 지원해야 한다. 최선형 IP 네트워크에 기초하는 많은 QoS 방법들이 제안되었으나, 이를 인터넷에 적용하기 위해서는 여러 네트워크 관리 영역들의 합의에 따라 진행할 수밖에 없는 실정이다.

실증적인 미래인터넷 연구를 위해서는 미국의 GENI, 유럽의 FIRE, 일본의 CoreLab과 같은 테스트베드가 필요하다. 최근 컴퓨팅과 네트워킹 성능이 비약적으로 발전하고 가상화 기술이 성숙해지면서, 독립적인 네트워킹 환경을 구성하고 여기에 실험적인 네트워킹 프로토콜들을 적용할 수 있는 미래인터넷의 초기 개념들이 현실화되고 있다. 이러한 추세에 부합하여 우리는 미래인터넷을 통해 미디어 지향 서비스 합성에 필요한 네트워킹 부분을 보다 효과적으로 개선할 수 있을 것으로 예상하고 있다. 미디어 중심 서비스 지향 테스트베드를 구축하기 위해서는 PC 기반으로 가상화를 지원하면서 프로그래밍이 가능한 네트워킹과 컴퓨팅 능력을 통합적으로 지원하는 노드들을 구성하고, 이를 활용하여 유무선 및 미디어 중심의 서비스들에 대한 동적인 합성(composition)을 지원해야 한다. 또한 다수의 실험들이 서로에게 어떤 간섭

도 없이 동시에 이루어지면서 개별 사용자들이 원하는 서비스들을 제공하기 위한 환경을 준비해야 한다.

따라서 본 고에서는 미래에 필요한 서비스들을 개발하고 적용해보는 소규모의 미디어 중심 서비스 지향 FIRST@PC (Future Internet Research for Sustainable Testbed @PC) 테스트베드를 제안한다. 구체적으로 설명하면, NetFPGA [2], OpenFlow [3]와 같은 네트워크 가상화 기술들을 활용하여 기초적인 미래인터넷 테스트베드를 구축할 수 있는 PC 기반의 프로그래머블/가상화 테스트베드 플랫폼 구조에 따라, 노드들을 구축하고 그 위에서 서비스를 제어하기 위한 에이전트 기반 소프트웨어를 운용한다. 최종적으로, 구축된 소규모 테스트베드 상에서 HD 미디어 분배를 위한 서비스 합성 실험을 수행하여 테스트베드의 실행 가능성을 탐색한다. 본 고의 구성은 다음과 같다. II절에서 미디어 중심 서비스 지향 테스트베드의 요구사항을 제시하고, III절에서 제안하는 테스트베드 플랫폼을 소개한다. IV절에서 구축된 프로토타입 테스트베드 및 이를 이용한 시범적인 검증 결과를 설명한 후, V절에서 본 고를 마무리한다.

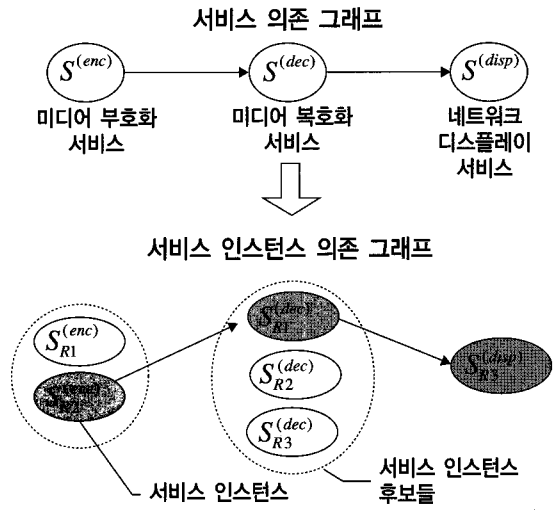
II. 미디어 중심의 서비스 지향 테스트베드의 정의 및 요구사항

본 절에서는 미디어 중심 서비스 합성에 대해 소개한 후, 서비스 지향 합성과 관련된 연구들에 대해 논의한다. 이후 서비스 지향 테스트베드가 고려해야 할 요구사항들에 대해 설명한다.

1. 미디어 중심 서비스 합성

일반적으로 서비스는 특정한 소프트웨어나 하드웨어가 제공하는 단위 능력을 주어진 접근/제어 권한에 따른 표준화된 인터페이스를 통해 다른 개체들이 이용할 수 있도록 도와준다. 서비스 합성은 서비스 지향 구조에 따라 기존 서비스들을 발견하고, 통합한 후, 실행함으로써 맞춤형 복합 서비스들을 자유롭게 구성하는 방법론이다. 서비스 합성을 상위 수준에서 기술하기 위해 [4]에서 제시한 서비스들의 실행 의존관계를 유향선분으로 연결되는 서비스 쌍들로 이루어

진 (그림 1)과 같은 서비스 의존 그래프를 이용한다. 여기서 유향선분은 서비스들의 호출 순서를 가리킨다.



(그림 1) 미디어 중심 서비스 합성 사례

서비스는 대표 이름, 데이터 처리 기능, 데이터 교환을 위한 인터페이스들로 구성된다. 또한 서비스 수행을 위해 필요한 최소 자원들의 종류와 양이 선언되어야 하는데, 이는 서비스 품질이 공급될 자원의 양에 상당히 의존적이기 때문이다. 서비스는 용도에 따라 컴퓨팅 서비스와 네트워킹 서비스로 구분된다. 컴퓨팅 서비스는 CPU/GPU 능력과 카메라, 디스플레이 등을 이용하여 미디어 스트림을 처리하며, 대표적인 서비스로는 영상 부호화/복호화 서비스, 트랜스코딩 서비스, 영상 분할 서비스 등이 있다. 네트워킹 서비스는 가상화된 네트워킹 능력을 활용하여 컴퓨팅 서비스들 간에 가공된 데이터를 연속적이고 신뢰성있게 전달하며, 대표적인 사례로는 데이터 전송 서비스, 멀티캐스팅 서비스, 방화벽/NAT 우회 서비스 등이 있다.

서비스 의존 그래프에 의해 선택된 서비스들은 테스트베드의 준비된 자원 위에 배치되고, 서비스 인터페이스들이 서로 연결됨으로써 합성 서비스를 구성하게 된다. 예를 들어, (그림 1)에서 보여주는 바와 같이, 미디어 부호화 서비스 $S^{(enc)}$ 는 두 개의 자원 중에서 하나를 선택한 후, 그 위에서 서비스 $S_{R2}^{(dec)}$ 를 실행시킬 수 있는데, 이때 선택된 자원 위에서 실행된 서비스를 서비스 인스턴스라고 부른다. 또한 연결된

서비스 인스턴스들의 의존 관계는 서비스 인스턴스 의존 그래프로 나타낸다.

2. 관련 연구

본 절에서는 미래인터넷에 적용가능한 서비스 지향 합성 연구들을 소개한다. 먼저 SILO (Service Integration, control and Optimization)은 네트워킹 프로토콜 세부 스택들을 컴포넌트 형태로 추상화하여, 사용자 요구사항을 충족시키고 성능을 최적화하도록 네트워킹을 구성한다 [5]. SILO 구조는 먼저 (1) 세밀한 네트워킹 기능 블록들을 서비스(예, 패킷 전달, 종단간 흐름 제어, 패킷 단편화)로 만들고, (2) 네트워킹 기능 블록들을 조합하여 복합적인 네트워킹 작업을 수행할 수 있도록 지원하며, (3) 크로스레이어 원칙하에서 블록간 상호작용을 촉진하여 네트워킹 자원의 활용성을 높이도록 설계되어 있다. 이렇게 함으로써 가용 자원의 범위 내에서 응용의 요구사항을 만족시키기 위해 가장 적합한 기능 블록들을 선택하고 그들의 제어 변수들을 주의 깊게 조절할 수 있다.

SILO와는 달리, [6]에서 제안된 네트워킹 서비스 구조는 중단 시스템들이 수행해왔던 컴퓨팅/네트워킹 기능들을 라우터들이 직접 제공하는 새로운 방식을 제안한다. 이 구조에서 중단 시스템은 전달할 정보 자체에 대해서만 관심을 가지며, 이 정보를 효율적으로 전달하기 위해 가공하는 일은 네트워킹이 담당하도록 설계된다. 중단 시스템이 제시한 서비스 제어 요청에 따라 라우터의 제어 평면에 있는 서비스 제어기들은 자신이 관리하는 서비스 노드들에게 필요한 네트워킹 서비스(예, 혼잡제어, 캐싱, 멀티캐스트, 암호화, 트랜스코딩)들을 할당하고 연결하여, 데이터들이 준비된 서비스 노드들의 경로를 따라 처리되고 전달되도록 조정한다. 또한 중단 유저들이 네트워킹 서비스들을 선택적으로 합성하여 복합적인 네트워킹 서비스를 구성할 수 있도록 추상화된 서비스 소켓 API들도 제공한다.

SpoVNet (Spontaneous Virtual Networks)은 응용의 요구사항에 기초하여 오버레이 네트워킹을 생성하는 구조로써, 이중 네트워크 상에서 유연하고 적응적이며 즉각적인 응용/네트워크 지향 서비스들을 구성하는 것을 목표로 한다 [7]. 순수한 P2P 네트워크와 달리, SpoVNet은 크로스레이어 정보를 통해 네트워크 인프라의 능력을 구체적으로 반영한 오버

레이 네트워크를 구성한다. SpoVNet은 언더레이 추상화와 서비스 추상화를 제공한다. 언더레이 추상화는 Locator/ID 분리에 기초한 주소체계를 채택하고 네트워크 인프라의 다양한 특성(예, 품질, 이동성)들을 추상화하여 상위 수준에서 연결성을 제공하는 인터페이스들을 제공한다. 사용자는 이 인터페이스들을 통해 보안, QoS 요구사항을 반영하는 네트워킹 서비스들을 만들 수 있다.

3. 미디어 중심 서비스 지향 테스트베드 요구사항

미디어 지향 서비스 합성 실험의 요구사항들은 실험 프로파일과 수준에 따라 여러 단계로 분류될 수 있다 [8]. 프로파일은 테스트베드가 제공할 수 있는 네트워킹 능력의 지원범위에 따라 달라진다.

- 이상적인 네트워킹 프로파일: 서비스 합성 실험에 필요한 대역폭이 언제나 보장된다는 이상적인 가정 하에서, 실험자는 컴퓨팅 서비스들의 자원 요구조건을 맞추는데 초점을 맞출 수 있다. 네트워킹 서비스들을 제외한 컴퓨팅 서비스들의 요구조건에 맞는 컴퓨팅 자원 후보들 중에서 일부를 선택하여, 그 위에 서비스들을 배치하고 연결한다.
- 실제적인 네트워킹 프로파일: 이 프로파일에서는 최선형 네트워크의 성능 변화를 관찰하면서, 컴퓨팅 능력도 함께 고려한다. 제공된 컴퓨팅 자원의 적정성과 프로그램래블 네트워킹 경로의 QoS 적합성을 모두 검증한 후 컴퓨팅 서비스들을 어디에 배치할 지를 신중히 결정할 수 있다.
- 가상화된 네트워킹 프로파일: 이 프로파일에서는 가상화된 네트워킹 인프라의 도움을 받아 네트워킹 서비스들을 실제로 만들 수 있다고 가정한다. 가상화된 네트워크 인프라가 제공하는 품질 보장형 네트워킹 경로들을 이용하여, 컴퓨팅 서비스들 간에 데이터 흐름을 연속적으로 보장한다. 컴퓨팅 서비스 인스턴스들은 다양한 수준의 QoS를 지원하는 네트워킹 서비스 인스턴스 후보들과 협상하여 가장 적합한 것들을 선택한다.

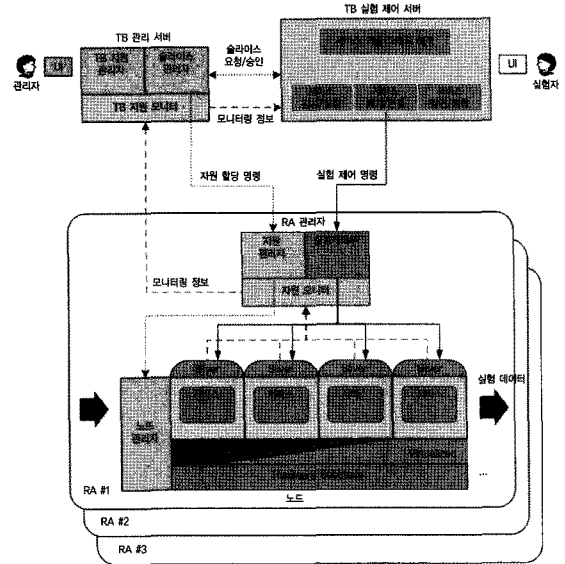
다음으로 실험 수준은 서비스 합성 실험들을 위한 운용 복잡도를 가리킨다. 미디어 지향 서비스 합성의 실험 수준에 따라 서비스 발견, 서비스 정합, 서비스 배치, 서비스 연결, 서비스 조절의 난이도가 다르게 적용될 수 있다.

- 레벨 0. 정적인 서비스 의존 그래프에 따른 고정적 서비스 배치: 이 수준에서는 서비스 의존 그래프가 실험 초기에 서비스 인스턴스 의존 그래프로 변형된 후 고정된다. 그러나 서비스와 자원으로부터 수집된 모니터링 결과에 따라 서비스들을 개별적으로 조절할 수 있다. 이 수준에는 컴퓨팅/네트워크 서비스들의 성능 변화를 고려할 필요가 적은 웹 서비스 합성 실험이 포함된다.
- 레벨 1. 조건부 서비스 의존 그래프에 따른 규칙 기반 적응형 서비스 배치: 이 수준에서는 조건부 분기를 이용하여 서비스 의존 그래프를 그릴 수 있으며, 주어진 상황에 따라 서로 다른 서비스 인스턴스 의존 그래프를 만들 수 있다. 또한 규칙에 따른 서비스 재배포를 할 수 있으므로, 상황에 따라 서비스 인스턴스를 다른 서비스 인스턴스로 바꿀 수 있다.
- 레벨 2. 초기 자율형 서비스 의존 그래프에 따른 규칙 기반 적응형 서비스 배치: 강화학습과 다중 에이전트 시스템과 같은 인공지능 기법들을 이용하여 서비스 의존 그래프를 자율적으로 계획할 수 있다. 실험자의 요구조건에 따라 서비스 의존 그래프들을 실험 초기에 자동적으로 만들지만, 실험 중에는 고정된 채로 남아있다. 그러나 변화된 환경에 맞게 서비스 인스턴스들은 적응적으로 재배포하는 것은 가능하다.
- 레벨 3. 자율형 서비스 의존 그래프에 따른 동적 서비스 배치: 이 수준에서는 자동적 그리고 반복적으로 서비스 의존 그래프를 생성/변경할 수 있다. 만약 합성된 서비스들에 문제가 발생할 경우, 보완된 대체 서비스 의존 그래프를 찾아 서비스 인스턴스들을 자동적으로 재설정할 수 있다. 지속적인 합성을 유지하기 위해서는 자원 가용성의 변화와 품질 저하의 가능성을 예측하여 서비스들과 자원의 연결관계를 재설정하는 등의 선제적인 대처가 요구된다.

III. FIRST@PC 미디어 중심 서비스 지향 테스트베드 플랫폼

이 절에서는 FIRST@PC 테스트베드 플랫폼을 소개한다.

(그림 2)에서 제안하는 테스트베드(TB) 플랫폼의 구성요소는 아직 설계 단계에 있으며, 부분적으로 구현된 상태이다.



(그림 2) FIRST@PC 테스트베드 플랫폼 구조

1. TB 관리 서버

TB관리 서버는 테스트베드의 모든 자원을 관리 및 동작 상태를 파악하고, 미디어 서비스 실험을 지원하기 위해 자원을 할당/회수하며, 실험자를 인증하고 사용권한을 부여한다. 현재 이 부분은 설계단계이며, 이를 현실화하기 위해 GENI 제어 프레임워크들과 연계하는 방안을 모색하고 있다. TB 관리 서버의 주요한 구성 요소는 실험에 소요될 가상화 자원들을 관리하는 슬라이스 관리자, 테스트베드 전체 자원을 관리하는 TB 자원 관리자, 그리고 테스트베드 자원들의 상태를 감시하는 TB 자원 모니터들로 구성된다.

슬라이스 관리자는 실험용 가상화 자원을 공급하는 요소이다. 공급중인 가상화 자원들에 대해 구체적으로 기록하고 있는 슬라이스 목록과 가상화 자원을 사용하고 있는 실험자에 대한 정보를 기록하는 사용자 목록들을 가지고 있다. 슬라이스 관리자는 RA(resource aggregate)들과 상호작용하여, 실험에 필요한 자원들을 할당하고, 상황에 따라 할당된 자원량을 탄력적으로 조정하며, 사용이 종료된 자원들을 회수한다. 부가적으로 슬라이스를 안정적으로 제공하기 위한

결합 포용 기능도 함께 제공한다.

TB 자원 관리자는 RA들을 테스트베드로 편입시킨 후 연동하는 행위를 돕는다. 임의의 RA가 테스트베드에 편입되려면, 기본적인 인증 절차를 거쳐 인가된 자원임을 확인받은 후, 이 RA가 가지고 있는 컴퓨팅/네트워킹 능력 정보를 TB 자원 관리자에게 제공해야 한다. TB 자원 관리자는 관리자 인터페이스를 통해 관리자에게 테스트베드 상태 정보를 보고하고, 필요한 테스트베드 제어 인터페이스들을 제공한다.

TB 자원 모니터는 RA들이 제공하는 자원들에 대한 활용 상태를 감시한다. 테스트베드 자원들이 정상적으로 동작하고 있는지, 어느 정도의 성능을 보여주고 있는지, 어떤 실험자에 의해서 얼마나 많이 사용되고 있는지를 세밀히 감시한다.

2. TB 실험 제어 서버

TB 실험 제어 서버는 서비스 합성 실험을 제어하는 서버이다. TB 관리 서버와 교신하여 획득한 가상 자원을 이용하여, 실험자가 제시한 실험 방식에 따라 가상 자원들을 설정하고, 소프트웨어를 설치한 후, 네트워크로 연결한다. 실험이 올바르게 수행되고 있는지 점검하기 위한 실험 평가 기능도 함께 제공한다. TB 실험 제어 서버는 GUI를 통해 에이전트 기반 서비스 제어 및 운용 기능을 제공하는 OMX (open media experiment) 소프트웨어를 운용한다. OMX 소프트웨어는 서비스 의존 그래프에 따라 필요한 서비스들을 발견한 뒤 자원과 정합시키고, 자원에 배치하여 연결시킨 다음, 서비스 합성 품질을 감시하고 제어하는 역할을 한다.

TB 실험 제어 서버는 실험자로부터 입력받은 서비스 의존 그래프를 해석한 후, 이 그래프에 포함된 서비스들의 소프트웨어 패키지를 저장하고 있는 서비스 저장소를 찾는다. 소프트웨어 패키지들은 특정 서버에 저장되어 있는 것이 일반적이거나, 각 자원들이 자체적으로 가지고 있는 경우도 있다. 필요한 서비스들에 대한 소프트웨어 다운로드가 모두 가능하다고 판별되면, 서비스 배치를 위한 정합 알고리즘을 이용하여 여러 후보 자원들 중에서 가장 적합한 자원들을 선택한다. 서비스 정합의 목표는 실험마다 다를 수 있으나, 개별 서비스들에 대한 품질 요구사항을 최대화하면서, 서비스 합성의 지속성을 안정적으로 유지하도록 정합하는 것이 중요하다.

TB 실험 제어 서버는 정합 결과에 따라 해당 서비스들을 선택된 자원으로 다운로드하여 설치한 후 실행한다. 이 과정을 통해 실행된 서비스는 서비스 인스턴스가 된다. 서비스 연결을 위해서, TB 실험 제어 서버는 서비스 의존 그래프에서 정의한 실행 순서에 따라 서비스 인스턴스들에게 이웃한 서비스 인스턴스들의 주소를 알려준다. 이 주소를 통해 각 서비스 인스턴스들은 그들의 인터페이스들을 실제로 연결하여, 최종적으로 서비스들을 통합한다.

TB 실험 제어 서버는 합성된 서비스들이 정상적으로 동작하는지 감시하며, 발생된 오류들을 실험자에게 즉각적으로 통지한다. 발생된 오류에 대처하기 위해, 실험자는 정의된 규칙에 따라 서비스들을 재설정하여 서비스 합성의 지속성을 유지할 수 있다.

3. Resource Aggregate

RA는 RA 관리자와 자원을 제공하는 노드들로 구성된다. RA 관리자는 노드를 관리하는 자원 관리자, 노드 성능을 감시하는 자원 모니터, 실험 제어를 위한 실험 제어부로 구성된다. 노드는 PC 기반의 컴퓨팅/네트워킹 장치로써, CPU/GPU, 메모리, 유무선 네트워크 인터페이스 등을 갖추고 있다. RA의 종류로는 미디어 처리 능력이 강조된 MediaX RA와 프로그래머블 네트워킹 능력에 초점을 맞춘 NetOpen RA가 있다.

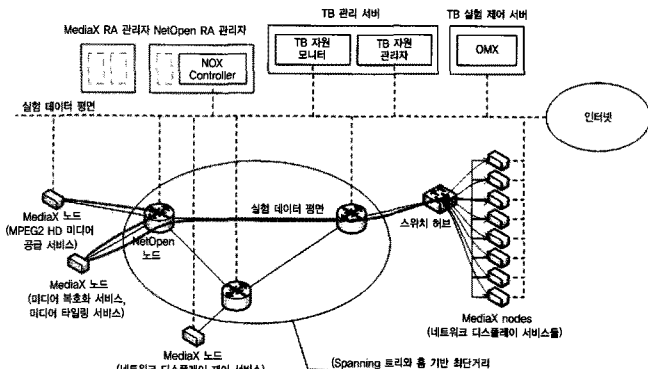
MediaX 노드는 카메라, 디스플레이 장치, 대용량 저장소, GPU 기반 고성능 워크스테이션과 같이 실시간, 대용량 미디어 처리에 적합한 컴퓨팅 자원들로 만들어진다. MediaX 노드는 가상화 수준에 따라 3가지 종류로 나눌 수 있다. 첫째, 가상화를 하지 않은 일반적인 MediaX 노드는 GPU 기반 그래픽스 가속 능력, 비디오 캡처 카드, 디스플레이와 같은 I/O 작업을 수행하는데 적합하다. 둘째, 가상화된 MediaX 노드는 주로 컴퓨팅 능력, 데이터 저장 능력을 필요로 하는 작업에 적합하다. 셋째, 클라우드 MediaX 노드는 분산된 컴퓨팅 자원들로 구성된 대규모의 가상 컴퓨팅 자원을 필요로 하는 작업에 활용될 수 있다.

NetOpen RA는 프로그래머블 네트워킹과 네트워크 가상화를 제공하는 OpenFlow 기반 PC 자원들의 집합이다. NetOpen RA 관리자는 NetOpen 노드를 제어하며, 실험자들이 네트워크 제어 도구 [9]를 통해 NetOpen 노드들을 이용한

네트워킹 실험을 할 수 있도록 도와준다. 플로우 제어를 위해, NetOpen 노드는 실험자가 조작 가능한 플로우 테이블을 제공한다. NetOpen 노드는 패킷 헤더를 분석하여, 플로우 별로 패킷들을 분류하고, 패킷들을 지정된 포트를 통해 전달할 수 있다. 네트워크 가상화는 NetOpen RA 관리자와 NetOpen 노드들 사이에 위치한 FlowVisor [10]를 통해 이루어진다.

IV. 시범적인 테스트베드 구축 및 검증

이 절에서는 현재까지 구축된 FIRST@PC 테스트베드 프로토타입에 대해 소개하고, HD 미디어의 분배 및 재생 서비스 합성 시연을 통해 테스트베드의 활용성을 탐색한다.



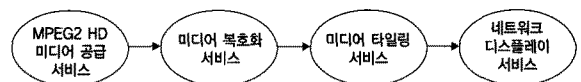
(그림 3) FIRST@PC 테스트베드 프로토타입 구성도

(그림 3)에서 나타난 바와 같이, 시범적으로 구축된 FIRST@PC 테스트베드 프로토타입은 노드들을 제어하기 위한 실험 제어 평면과 노드들 간에 데이터 전송을 위한 실험 데이터 평면으로 구성된다. 또한 테스트베드를 제어하기 위해 TB 관리 서버, TB 실험 제어 서버, RA 관리자들을 포함한다. TB 관리 서버는 테스트베드 자원 관리를 위해서 OMF를 운용하고, Ganglia 모니터링 툴을 이용하여 노드들의 성능 변화를 관찰한다. TB 실험 제어 서버에서는 미디어 지향 서비스 합성 실험을 제어하기 위해 OMX 소프트웨어를 설치/운용한다. NetOpen RA 관리자는 실험 제어 부분에 한정하

여 NOX Controller를 운용하며, MediaX RA 관리자는 현재 구현되지 않았다.

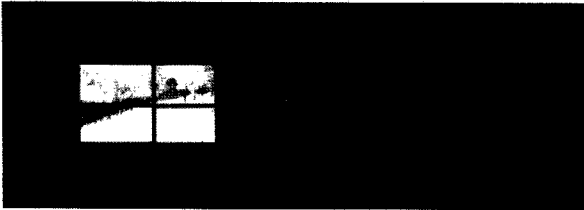
MediaX 노드는 크게 세 가지로 구성된다. 미디어 서버는 HD 720p/1080i 미디어를 공급하고, 미디어 변환기는 GPU의 도움을 받아 미디어를 고속 처리할 수 있으며, 디스플레이는 8대의 40인치 디스플레이들을 격자형으로 배열하여 초고해상도 네트워크 디스플레이 기능을 제공한다. MediaX 노드는 이더넷 인터페이스를 통해 NetOpen 노드와 연결된다. NetOpen 이 제공하는 실험 데이터 평면은 1Gbps 이더넷 인터페이스를 통해 2계층 수준의 연결성을 제공하며, 지역 네트워크의 범위를 넘어서는 경우에는 이더넷/IP 터널링을 통해 상호 연결성을 제공한다.

실제적인 네트워킹 프로파일에서 레벨 0 (정적인 서비스 의존 그래프에 따른 고정적 서비스 배치)에 대응하는 실험 시나리오에 따라 (그림 4)와 같이 서비스 의존 그래프를 구성하였다. MPEG2 HD 미디어 공급 서비스는 압축된 영상을 미디어 복호화 서비스로 보낸다. 미디어 복호화 서비스는 압축된 영상을 복원하고, 미디어 타일링(tiling) 서비스를 통해 네트워크 디스플레이의 각 타일 크기에 맞게 영상을 분할한 후, 네트워크 디스플레이 서비스에 전송한다. 위 과정에서 서비스들을 연결하는 모든 유향 선분들은 네트워킹 서비스를 대표한다. 여기서 네트워킹 서비스는 홉 기반 최단거리 경로를 구성하는 라우팅 기능을 제공한다.



(그림 4) 서비스 의존 그래프

(그림 3)에서 적색 표기된 화살표 선들은 (실험자의 판단에 따라) 배치된 서비스 인스턴스들의 연결 상태와 트래픽 흐름을 보여준다. 이와 같은 합성 실험의 결과에 따라 (그림 5)와 같이 네트워크 디스플레이 상에서 HD 영상으로 표현된다.



(그림 5) 네트워크 디스플레이상의 HD 영상 표현

V. 결론 및 향후 연구

본 고에서는 컴퓨팅/네트워킹 자원들을 이용하여, 서비스들을 통합적으로 운용 및 제어하는 서비스 지향 테스트베드를 소개하였다. 또한 서비스 제어 및 운용 소프트웨어를 이용하여 HD 미디어 분배 및 재생 실험을 통해, 기본적인 미디어 지향 서비스 합성을 실험적으로 검증할 수 있음을 보였다. 향후에는 KREONET/KOREN 연구망을 통해, 소규모 테스트베드를 외부의 여러 노드들과 연결하여 그 규모를 확장하면서, 여러가지 창의적인 서비스들을 개발하고 실험할 예정이다.

Acknowledgement

본 논문은 “미래인터넷 인프라를 위한 가상화 지원 프로그램머블 플랫폼 및 핵심원천 기술개발” 과제 (2009-F-050-01)에 대한 결과물중 일부분임.

참 고 문 헌

- [1] K. Nahrstedt and W.-T. Balke, “Towards building large scale multimedia systems and applications: challenges and status,” in Proc. of ACM Int. Workshop on Multimedia Service Composition, pp. 3-10, Nov. 2005.
- [2] G. Gibb, J.W. Lockwood, J. Naous, P. Hartke, and N. McKeown, “NetFPGA-An open platform for teaching how to build Gigabit-rate network switches and routers,” IEEE Trans. on Education, vol. 51, no. 3, pp. 364-369, Aug. 2008.
- [3] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling innovation in campus networks,” ACM SIGCOMM Computer Communications Review, vol. 38, no. 2, pp. 69-74, Mar. 2008.
- [4] X. Gu and K. Nahrstedt, “Distributed multimedia service composition with statistical QoS assurances,” IEEE Trans. on Multimedia, vol. 8, no. 1, pp. 141-151, Feb. 2006.
- [5] R. Dutta, G.N. Rouskas, and I. Baldine, A. Bragg, and D. Stevenson, “The SILO architecture for services integration, control, and optimization for the Future Internet,” in Proc. of IEEE ICC, pp. 1899-1904, Jun. 2007.
- [6] S. Shanbhag and T. Wolf, “Implementation of end-to-end abstractions in a network service architecture,” in Proc. of ACM CoNEXT, pp. 1-12, Dec. 2008.
- [7] SpoVNet Consortium, “SpoVNet: an architecture for supporting Future Internet applications,” in Proc. of EuroView, Jul 2007.
- [8] S. W. Han and J. Kim, “Preparing experiments with media-oriented service composition for Future Internet,” in Proc. of Int. Conf. on Future Internet Technologies, pp. 73-78, Jun. 2010.
- [9] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, “NOX: Towards an operating system for networks,” ACM SIGCOMM Computer Communications Review, vol. 38, no. 3, pp. 105-110, Jul. 2008.
- [10] R. Sherwood et al., “Carving research slices out of your production networks with OpenFlow,” ACM SIGCOMM Computer Communications Review, vol. 40, no. 1, pp. 129-130, Jan. 2010.

약 력



한 상 우

2003년 중앙대학교 컴퓨터공학과 학사
 2005년 광주과학기술원 정보통신공학과 석사
 2005년 - 현재 광주과학기술원 정보통신공학부 박사과정
 관심분야: 멀티미디어 서비스 합성, 미래인터넷 테스트베드,
 서비스 지향 컴퓨팅, 다자간 협업 환경



김 남 곤

2004년 전남대학교 정보통신공학부 학사
 2006년 광주과학기술원 정보통신공학과 석사
 2006년 - 현재 광주과학기술원 정보통신공학부 박사과정
 관심분야: 프로그래머블 네트워크, 네트워킹 서비스,
 미래인터넷 기술



차 병 려

1995년 호남대학교 수학과 학사
 1997년 호남대학교 컴퓨터공학과 석사
 2004년 목포대학교 컴퓨터공학과 박사
 2005년 - 2009년 호남대학교 컴퓨터공학과 전임강사
 2009년 - 현재 광주과학기술원 SCEIT 연구센터 연구교수
 관심분야: 정보보안, 신경망 학습, 미래 인터넷 기술



김 중 원

1987년 서울대학교 제어계측공학과 학사
 1989년 서울대학교 제어계측공학과 석사
 1994년 서울대학교 제어계측공학과 박사
 1994년 - 1999년 공주대학교 전자공학과 조교수
 1998년 - 2001년 미국 Univ. of Southern California, Los Angeles,
 CA, EE-System Department 연구조교수
 2000년 - 2001년 미국 InterVideo Inc., Fremont, CA, 개발 자문
 2001년 - 현재 광주과학기술원 정보통신공학부 교수
 관심분야: Networked Media Systems and Protocols focusing
 "Dynamic Composition of Immersive Media-
 oriented Services over the Wire/Wireless IP
 Convergence Networks".

