主題

DiffServ를 이용한 인터넷 QoS 보장 기술

대구가톨릭대학교 전용희, 박수영

차 례

Ⅰ. 서 론
Ⅱ. DiffServ 구조 모델
Ⅲ. 홉별 행동(PHB)
Ⅳ. 스케줄링 알고리즘
Ⅴ. 대역폭 브로커(BB: Bandwidth Broker) 시스템
Ⅵ. DiffServ와 ATM
Ⅶ. 결 론

Ⅰ. 서 론

현재의 인터넷은 최선 노력(Best Effort) 서비스만을 지원하고 있으나, 인터넷 전화, 인터넷 방송, VPN(Virtual Private Networks), 멀티미디어 서비스 등의 실시간 혹은 높은 대역폭 요구 서비스들이 늘어남에 따라 인터넷에서도 서비스 품질(QoS: Quality of Service)에 대한 요구가 큰 문제가 되고 있다.

QoS 보장을 위한 손쉬운 접근 방법은 폭주가 일어나지 않도록 백본의 대역폭을 충분히 크게 확보하는 것이다. 그러나 대역폭만 크게 하고 현재의 최선 노력 서비스 구조를 그대로 유지할 경우 QoS가 보장될 수 없다. 현재의 서비스 구조에서는 특정 노드나 네트워크에서 불시에 폭주가 발생할 수 있는데, 이런 폭주를 모두 예상해서 백본의 대역폭을 높인다는 것은 비경제적이기도 하지만 불가능한 일이다. 따라서 대역폭을 충분히 크게 해도 최소한의 QoS 보장방법이 수반되어야만 원하는 성능을 얻을 수 있다[1].

인터넷의 표준화 기구인 IETF에서는 현재 인터넷상에서 서비스 품질을 보장하기 위한 방법들에 대하여 표준화 작업을 진행하고 있다. 그 중에서 대표적인 것이 통합 서비스(IntServ: Integrated Services)와 차별화 서비스(DiffServ:Differen-tiated Services) 이다. 통합 서비스는 RSVP (Resource Reservation Protocol)라는 신호 프로토콜을 이용하여 자원을 할당한다. 그러나, 통합 서비스 모델에서는 특정한 패킷 스트림 혹은 플로우(flow)에 대하여 자원을 할당하기 때문에, 연결 상태에 관련된 정보를 저장하고 처리하는 것이 문제가 된다. 특히, 인터넷 백본 라우터의 경우 전송속도가 빠르고 연결된 플로우의 수가 많기 때문에 통합 서비스 모델을 지원하기 힘든 확장성(scalability) 문제가 발생한다.

DiffServ[2 3]에서는 흐름별로 트래픽을 관리하

지 않고 패킷을 클래스별(집합)로 분류해서 처리하기 때문에 RSVP와 같은 확장성의 문제가 없다. 본 논문에서는 DiffServ(DS)를 이용하여 QoS를 보장할 수 있는 DS 망 및 DS 라우터의 구성요소에 대해서 알아보고, 차별 서비스를 지원하는 DiffServ 프레임워크에 대하여 기술하고자 한다.

Ⅱ. DiffServ 구조 모델

DiffServ 모델에서는 트래픽을 플로우별로 다루지 않고 서비스 클래스로 분류하여 처리함으로써 중간 라우터에서 고속의 패킷 전달을 할 수 있도록 하는 것을 기본으로 한다. 이를 위하여, DiffServ 구조에서는 망에 들어가는 트래픽을 망의 경계에서 분류하고 조절하며, 다른 BA(Behavior Aggregates)에 할당한다. 각 BA는 한 개의 코드 포인터(DSCP: DS Code Point)에 의하여 표시되며, 망의 코어 내에서는 이 DSCP와 관련된 PHB(Per Hop Behavior)에 따라 패킷이 전달된다.

1. 차별 서비스 영역

차별 서비스 영역(DS Domain)은 공통의 서비스 제공 정책과 각 노드에서 구현된 PHB 그룹의 집합으로 동작되는 DS 노드의 인접된 집합을 말한다. DS 영역은 DS 경계 노드들로 이루어진 잘 정의된 경계를 가지며, 여기에서 영역을 통과하는 패킷들이

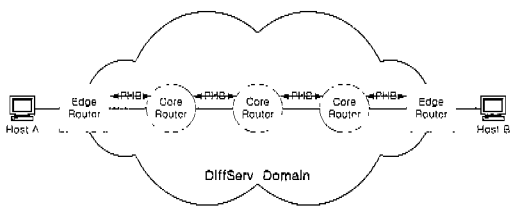

그림 1. 단일 DS 영역

영역 내에서 지원되는 PHB 그룹 중 한 개의 PHB를 선택하기 위하여 적절히 마크되도록 입력 트래픽을 분류하고 조절할 수 있다. DS 영역 내의 노드는 DSCP를 기초로 패킷에 대한 전달 행동을 선택하게 된다. 그림 1은 DS 단일 영역을 보여준다.

그림 1에서 DS 영역은 경계노드(Edge Router)와 내부 노드(Core Router)로 구성된다. DS 경계 노드와 내부 노드는 둘 다 DSCP에 기초하여 패킷에 적절한 PHB를 적용할 수 있어야 한다. DS 경계노드는 복잡한 분류와 DSCP를 기초로 패킷에 대하여 적합한 PHB를 적용하는 트래픽 조절 기능이 구현되도록 하고, 반면 내부 노드는 PHB에 해당하는 적절한 행동으로 트래픽 전달 처리를 수행한다. 이와 같이 DiffServ 특징 중의 하나가 에지(edge)와 코어(core)를 구분하는 것이다. 즉, 트래픽 분류, 폴리싱(policing) 등의 기능을 에지 부분에 할당하고 대신에 코어 부분에서는 패킷 전달을 단순화시킨다.

그림 1의 DS 망은 RSVP에 의한 자원예약의 번거로움 및 복잡성을 피하기 위해 한 홉에서 다음 홉 사이에 패킷이 처리되어야 하는 PHB를 규정하고 있다. 이 PHB는 홉 단위의 패킷 처리 행동 양식을 말한다. 다음은 차등 서비스의 개략적인 행동에 대한 설명이다.

① 호스트 A와 호스트 B간 서비스의 품질을 보장받기 위해서 호스트 A는 인접한 서비스 제공자와 계약을 맺게 된다. 여기서의 계약은 호스트 A로부터 전송되는 트래픽에 대한 서비스 품질의 정도를 규정할 수 있다. 이러한 계약이 이루어진 다음 호스트 A는 차등 서비스망의 에지 라우터로 패킷을 전송하게 된다.

② 에지 라우터로 입력된 호스트 A의 트래픽에 대해 에지 라우터는 계약된 서비스의 품질을 보장하기 위한 동작을 수행한다. 우선 입력된 패킷이 어느 호스트로부터 입력된 것인지 분류되고 이렇게 분류된 호스트 A의 트래픽이 계약을 위반하는지 여부를

판단한 다음, 지역 정책에 따라 해당하는 처리를 한다. 이러한 과정을 거친 트래픽은 차등 서비스 망 내부에서의 패킷 처리 규칙에 해당하는 PHB로 변환되어야 한다. DS 망 내에서는 PHB에 따라 패킷을 처리한다. 따라서 각 패킷이 어떤 PHB에 속하는지, 즉 어떠한 패킷 처리를 받아야 하는지에 대한 판단을 해야하는데 이러한 판단 기준으로 IP 패킷 헤더 내의 DSCP 값이 사용된다. 따라서 차등 서비스 망의 에지 라우터는 입력되는 패킷에 따라 계약에 맞는 적절한 패킷 처리를 받을 수 있도록 적절한 DSCP 값을 인코딩 해야한다. 같은 값으로 인코딩된 패킷들은 동일한 패킷 처리 규칙을 따라야 하므로 동일한 큐에 큐잉되고 처리된 후 코어 라우터로 전달된다.

③ 코어 라우터는 입력된 패킷의 DSCP 값을 검사하여 어떤 PHB에 속하는지 결정하고 그 PHB에 합당한 처리 규칙에 따라 패킷을 처리한 후 다음 홉으로 전달한다. 이렇게 코어 라우터는 단순히 패킷의 DSCP값만을 보고 패킷을 처리하므로 각 플로우의 상태를 알 필요가 없게되고, 또한 각 플로우별로 분류하여 처리해 줄 필요가 없어지므로 간결한 동작으로 패킷을 다음 홉으로 전달할 수 있다.

④ 이런 과정을 거쳐 호스트 A의 데이터는 호스트 B까지 요청된 QoS를 보장받으면서 전송된다.

이처럼 코어 라우터에서의 패킷 처리가 단순해지게 된 것은 홉간의 패킷 처리 규칙을 규정한 PHB 개념의 도입으로 가능해졌다. 차등 서비스에서 PHB는 한 라우터에서 다른 라우터로 트래픽을 전달하는 기본 방침이며 각 라우터에서 동일한 DSCP 값을 가지는 패킷들의 집합인 BA에 자원을 할당해 주는 방침으로 홉 단위의 자원할당 방식을 기본으로 하고 있다. PHB는 다른 PHB들 보다 상대적으로 버퍼나 대역폭에 대하여 우선 순위를 갖는 것으로 정의될 수 있고, 지연 한계나 손실률을 기술한 트래픽 특성으로도 정의될 수 있다.

2. 차별 서비스 지역

차별 서비스 지역(DS Region)은 한 개 이상의 인접 DS 도메인의 집합이다. DS 지역은 지역 내의 도메인을 연결하는 경로를 따라 차별 서비스를 지원할 수 있다. DS 지역 내의 DS 영역은 내부적으로 다른 PHB 그룹과 다른 코드포인트와 PHB 매핑을 지원할 수 있다. 그러나 영역간에 걸친 서비스를 허용하기 위하여, 패킷을 보내는 영역과 받는 영역사이에서 계약이 필요하다. 이와 같이 두 개의 DS 도메인간의 경계에서 조절되는 방법을 명시하는 것을 TCA(Traffic Conditioning Agreement)라 하며, 이러한 형태의 계약을 SLA(Service Level Agreement)라고 한다[4](그림 2 참조).

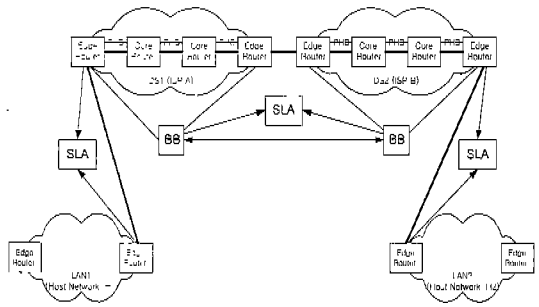

그림 2. DS 지역(Region)

그림 2에서는 서비스 계약이 BB(Bandwidth Broker)라고 하는 대리인(agent)을 통하여 이루어지는 것을 보여준다. BB는 DS 구조에서 적당한 자원 관리를 위하여, 이용할 수 있는 자원과 입력 SLA에 의하여 자원을 할당하는 중앙 에이전트라고 볼 수 있다. BB는 SLA에 의하여 할당된 대역폭에 관한 정보를 저장하고, 향후 할당을 결정하기 위하여 기초로 하는 정책(policy) 데이터 베이스를 가진 소프트웨어라고 볼 수 있다. BB에 대한 보다 자세한 내용은 뒤에 기술되었다.

3. 기능 블록(Functional Blocks)

네트워크에 진입하는 트래픽을 조절하기 위하여 그림 3에서 보여주는 바와 같이 다음의 기능들이 에지 부분에 할당된다: 분류기, 미터, 마커, 쉐이퍼. 그러나, 4가지 모든 요소를 반드시 포함할 필요는 없다. 예를 들어, Null 미터인 경우, 패킷이 분류기에서 마커나 쉐이퍼로 직접 전달될 수도 있다.

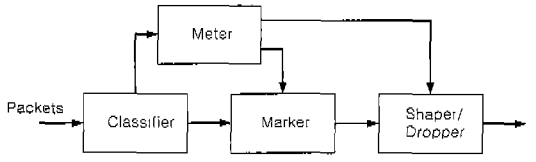

그림 3. 기능 블록의 논리적 관점

(1) 분류기(Classifier)

패킷 분류기는 패킷 헤더의 내용을 기초로 트래픽 스트림에 있는 패킷을 선택한다. 분류기에서 사용하는 분류방법으로 두 가지가 있다. 하나는 BA (Behavior Aggregate) 분류기로 DS 필드만을 기초로 패킷을 분류한다. 다른 방법은 MF (Multi-Field) 분류기로 근원지 주소, 목적지 주소, DS 필드, 프로토콜 ID, 근원지/목적지 포트 (TCP, UDP)와 같은 전송 계층 헤더 필드 등의 패킷 헤더에 있는 여러 필드의 조합으로 분류를 한다. 이와 같이 패킷의 플로우를 확인하는 과정을 분류라고 한다.

(2) 미터(Meter)

트래픽 미터는 TCA에서 명시된 트래픽 프로파일에 대해 분류기에 의해 선택된 패킷 스트림의 시간적 특성을 측정한다. 이 기능 블록은 패킷이 분류기에 의하여 분류된 다음 그 패킷이 소비하는 자원을 알기 위한 블록이다. 미터는 어떤 트래픽 흐름이 자원의 소비 제한 내에 있는지(이를 in-profile이라고 한다), 아니면 초과하는지(이를 out-of-profile이라고 한다)를 판단하기 위하여 각 패킷에 대하여 점사하고 그리고 특정한 행동을 개시하기 위하여 다른 조절 기능에 상태 정보를 전달한다. 패킷을 측정하는 파라미터로는 흐름 율(flow rate)과 버스트 크기가 있다.

미터의 유형으로 AR(Average Rate) 미터, EWMA(Exponential Weighted Moving Average) 미터, TB(Token Bucket) 미터 등이 있다[5].

(3) 마커(Marker)

비록 미터링과 마킹이 두 가지 기능으로 분리되어도, 거의 동일한 제안에서 설명된다. 미터는 패킷을 트래픽 프로파일과 비교하고, 마커는 미터의 결과에 따라 적합한 코드포인트로 패킷을 마킹하는 기능을 수행한다. 이와 같이, 마커는 패킷 헤더 내의 DSCP를 설정하며, DSCP 값이 0으로 제출된 마크되지 않은 패킷에 대하여 뿐만 아니라, 사전에 마크된 패킷에 대하여도 재 마킹을 할 수 있다. 패킷을 마킹하는 방법으로 두 가지의 방법이 있을 수 있다. 1) 두 비트 차별화(in-profile, out-of-profile로 표시)[6], 2) 세 비트 차별화(탈락 우선권이 low, medium, high로 표시)[7 8].

(4) 쉐이퍼(Shaper)

쉐이퍼는 트래픽 프로파일에 따라 스트림을 보내기 위해 트래픽 스트림에서 약간의 패킷이나 혹은 모든 패킷을 지연시킨다. 쉐이퍼는 일반적으로 유한 크기 버퍼를 가지고, 만약 지연된 패킷을 잡아두기에 충분한 버퍼 공간이 없다면 패킷은 탈락될 것이다. 이와 같이 버스트 데이터를 처리하거나 트래픽의 속도를 조절하는 것을 쉐이핑이라고 하며, 버스트로 도착하는 트래픽을 평활화(smooth)하기 위하여 사용될 수 있다.

(5) 탈락기(Dropper)

탈락기는 트래픽 프로파일에 따라 스트림을 보내기 위해 트래픽 스트림에서 약간의 패킷 혹은 모든 패킷을 탈락(drop)시킨다. 탈락기는 패킷이 없거나 적은 양의 패킷을 위해 쉐이퍼 버퍼 크기를 설정함으로서 쉐이퍼의 특별한 경우로 구현될 수 있다.

탈락기는 단시간의 버스트를 허용하면서 장시간의 폭주를 막기 위하여 능동적인 대기 관리 알고리즘, 예를 들어, RED(Random Early Detection)나 RIO(RED with In/Out) 등이 사용된다.

앞에서 기술한 미터/마커, 쉐이퍼, 탈락기의 기능을 종합하여 폴리싱(policing) 이라고 한다. 트래픽 조절기는 대개 DS 입구/출구 경계 노드 내에 위치하지만, DS 영역의 내부 노드나 non-DS 영역 내에 위치할 수도 있다.

Ⅲ. 홉별 행동(PHB)

1. PHB(Per Hop Behavior)의 개념

PHB는 어떤 특정한 DS 행위 집합(BA: Behavior Aggregates)에 적용되는 DS 노드의 외부적으로 관찰할 수 있는 전달 행위(FB: Forwarding Behavior)의 묘사이다[9 10]. PHB는 노드가 BA에 대하여 자원을 할당하는 수단이다. PHB는 다른 PHB에 비해 그들의 자원(예, 버퍼, 대역폭), 우선 순위 혹은 상대적인 관찰 가능한 트래픽 특성(예, 지연, 손실)에 의하여 명시될 수 있다. 이런 PHB들은 자원을 할당하기 위한 빌딩 블록으로 사용될 수 있으며, 일관성을 위해 PHB 그룹으로 명시되어야 한다. PHB 그룹은 통상적으로 패킷 스케줄링이나 버퍼 관리 정책과 같이, 그룹 내의 각 PHB에 적용되는 공통의 제약 조건을 공유한다. 그룹 내의 PHB 간의 관계는 절대적 혹은 상대적인 우선권에 관한 것일 수 있다. (예, 결정적 혹은 확률적 임계치 방법에 의한 탈락 우선권)

PHB는 어떤 버퍼 관리와 패킷 스케줄링 메커니즘에 의해 노드 내에서 구현된다[11]. PHB는 서비스 공급 정책에 관련한 행위 특성에 의하여 정의되며, 특정한 구현 메커니즘에 의한 것은 아니다. 일반적으로, 다양한 구현 메커니즘이 하나의 특정한 PHB 그룹 구현에 적합할 수도 있다. 게다가, 하나의 노드 상에 한 개 이상의 PHB 그룹이 구현되어 영역 내에서 사용될 수도 있다.

그림 4와 그림 5는 각각 IP헤더 내의 DSCP 필드의 위치와 DiffServ 필드의 정의를 나타낸다. PHB들은 IP헤더 내의 DSCP 필드에 해당 PHB를 나타내는 특정 값으로 인코딩 된다.

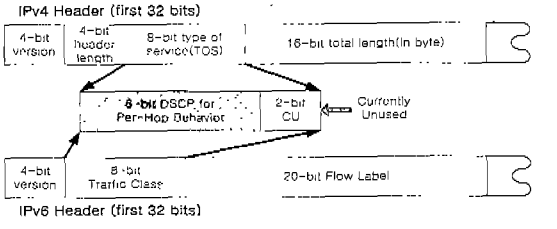

그림 4. IP 헤더 내의 DSCP 영역

IPv4와 IPv6 헤더의 ToS(Type of Service) 필드는 DSCP 필드(6 bit)와 미사용 필드(2 bit)로 나누어진다. DSCP 코드포인트는 3비트씩 나누어진다. 첫 번째 3 비트는 클래스 셀렉터(Class Selector) 코드포인트이고 트래픽 클래스를 나타낸다(그림 5 참조). 예를 들어, EF(Expedited Forwarding), AF(Assured Forwarding), 혹은 BE(Best Effort)를 나타낼 수 있다. 이 트래픽 클래스는 라우터에 의해 변경되지 않는다. 두 번째 3 비트는 탈락 우선권(Drop Precedence)으로 불려지는데 패킷을 초기에 탈락시킬지(high drop precedence), 나중에 탈락시킬지(low drop precedence)를 결정한다. 어떤 제안에서는

medium drop precedence로 알려진 세 번째 탈락 우선권도 있다. 트래픽은 단일 링크간의 차별화가 아니라 트래픽 클래스를 위한 링크 집합에서만 차별화한다.

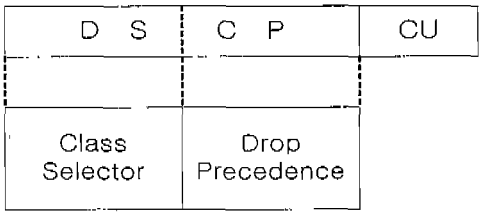

그림 5. DiffServ 필드 정의

2. 제안된 PHB의 종류

DS Working Group(DSWG)은 IPv4의 ToS 필드와 IPv6의 8-bit 클래스 비트를 재 정의하고 그것을 DS 필드라고 부른다. DSWG는 이종 PHB를 위해 사용된 DS 필드 공간으로부터 DSCP를 할당한다.

여러 인터넷 초안들(Internet Draft)은 이런 DSCP의 사용을 권고하고 있다. DSCP에는 Default PHB[12], Class Selector Compliant(CSC) PHB Group[12], EF PHB[8], AF PHB[7], Dynamic Rt/Nrt PHB Group[13]을 포함한다.

표 1은 AF/EF 코드포인트 충돌에 대해 제안된 해결책이다. 그러나 제안만 되었을 뿐 표준 코드포인트는 아니다. 표 1에서 제안된 코드포인트는 8개로 분리된 DSCP에 의해 표시되는 8개의 우선권이나 탈락 우선권(DP: Drop Precedence)을 포함하는 PHB를 지원한다[13]. 각 PHB를 위한 DSCP 값은 다음과 같다: DP0=000xxx, DP1=001xxx, DP2=010xxx, DP3=011 xxx, DP4=100xxx, DP5=101xxx, DP6=110 xxx, DP7=111xxx. 탈락 우선권이 낮은 레벨(DP0)은 상대적으로 높은 순서를 나타낸다.

표 1. 제안된 DS 코드포인트

DSCP	PHB	DSCP	PHB
000000	Default PHB, Class Selector-DP0	100000	Class Selector-DP4, AF31
000010	DRT_11	100010	AF32
000100	DRT_12	100100	AF33
000110	DRT_13	100110	DRT_23
001000	Class Selector-DP1	101000	Class Selector-DP5, AF41
001010	DRT_14	101010	AF42
001100	DRT_15	101100	AF43
001110	DRT_16	101110	EF
010000	Class Selector-DP2, AF11	110000	Class Selector-DP6
010010	AF12	110010	DRT_24
010100	AF13	110100	DRT_25
010110	DRT_21	110110	DRT_26
011000	Class Selector-DP3, AF21	111000	Class Selector-DP7
011010	AF22	111010	no recommendation
011100	AF23	111100	no recommendation
011110	DRT_22	111110	no recommendation

DiffServ 트래픽 차별화에서의 주요 개념은 상대적으로 높은 순서를 나타내는 DP 레벨로 마킹된 패킷이 상대적으로 낮은 순서를 나타내는 DP 레벨로 마킹된 패킷보다 적시에 전달되는 확률이 더 높다는 것을 보장하는 것이다.

DRT_1x는 비실시간 트래픽을 위한 것이고, DRT_2x는 실시간 트래픽을 위해 제안된 코드포인트이다. x는 탈락 우선권 값이다.

3. 표준화된 PHB의 종류

(1) Default(DE) PHB

특정 서비스를 요구하지 않는 패킷에 대해서는 기본적으로 DE로 서비스 해준다. 인터넷 라우터에서 취해지는 패킷 전달 방식인 최선 노력과 같은 정도의 서비스를 나타내며, 지연이나 손실 같은 보장된 전달 행동이 없는 IP 네트워크 상의 서비스로 알려져 있다. EF나 AF와 같이 명확한 코드포인트로 설정되지 않은 나머지 서비스는 DE 트래픽으로 전송된다.

이 PHB에 대한 서비스 수준은 다른 PHB의 출력 링크의 요구가 없을 때 서비스 해주는 것을 원칙으로 한다. 그러나 차등 서비스를 사용하지 않는 서비스를 위해서 항상 최소한의 자원을 할당해 놓아야 하고 반드시 수용되어질 수 있어야 한다. 이 방법에서 패킷은 순서대로 출력되고 손실 가능성이 있다. 지연은 가능한 최소화되고 대역폭은 가능한 많이 이용된다.

DE PHB에 속하는 트래픽은 분리된 큐에 큐잉되는 것 이외에 DiffServ 장점의 어느 것도 경험하지 못한다. 이런 큐에 패킷이 가득 채워질 때 패킷은 탈락된다. DE 트래픽에 해당하는 어플리케이션 예는 FTP와 웹 브라우징과 같은 전통적인 컴퓨터 데이터 전송이다.

코드포인트는 그림 6에서 보여준다.

0	0	0	0	0	0

그림 6. DE 코드포인트

(2) Expedited Forwarding(EF) PHB[8]

라우팅 정보 갱신과 같은 망 제어 트래픽 전달에 사용되는 우선 순위가 높은 전달 방식이다. 패킷은 작은 지연과 지연 변이, 적은 손실을 가지며 대역폭 보장 서비스를 받을 수 있다. 양질의 서비스를 요구하는 트래픽을 서비스하기에 적절한 PHB이다.

이런 서비스 보장을 위해서 트래픽이 큐잉되지 않거나 아주 적게 큐잉 되도록 보장해야 한다. 큐는 단기간(short-term) 트래픽 도착 율이 출발 율(departure rate)을 초과할 때 발생한다. EF 패킷이 실시간으로 전송되도록 보장하기 위해 EF PHB는 DiffServ 노드로부터 패킷 출발 율이 배치된(configured) 율과 같거나 초과하도록 정의한다.

표준 동작에서는 EF 클래스에 대해 폭주가 발생하는 일은 없다고 가정한다. 어떻게 해서든 EF 서비스 메커니즘을 초과하는 패킷은 탈락시킬 수 있다. EF PHB에 해당하는 패킷의 속도는 사전에 계약된 즉, 트래픽 프로파일에 정의되어있는 최대 사용 대역폭에 따라서 엄격하게 지켜져야 한다. 일반적으로 트래픽을 어기고 초과해서 들어오는 EF 트래픽에 대해서는 쉐이퍼(shaper)로 속도를 트래픽 프로파일에 정의된 속도로 조정하고 쉐이퍼 사용에도 불구하고 조절이 안 되는 트래픽에 대해서는 폐기(discard)를 한다.

이렇게 쉐이핑된 플로우는 엄격한 PQ(Priority Queueing)나 높은 가중치를 할당받는 WFQ (Weighted Fair Queueing) 등의 스케줄링 기법을 통해 엄격한 서비스를 제공받는다. 이러한 특성으로 인해 EF PHB에 속하는 트래픽은 가상 전용선 서비스(VLL: Virtual Leased Line)와 같은 서비스를 받을 수 있다. 그러므로 최대 비트율이나 고정된(constant) 비트율을 가진 실시간 어플리케이션을 위해 적합하다. EF를 위한 어플리케이션 예로는 Voice over IP, Video over IP 등이 있다.

EF PHB를 위한 표준 코드포인트는 그림 7과 같다. 코드포인트는 다른 PHB 그룹과 중복되지 않는다.

1	0	1	1	1	0

그림 7. EF 코드포인트

(3) Assured Forwarding(AF) PHB[7 14]

AF PHB 그룹은 고객 DS 도메인으로부터 수신되는 IP 패킷을 위해 다른 전달 보장 레벨을 제공하는 제공자를 위한 수단이다. AF PHB에 속하는 트래픽은 망의 혼잡 상황에서도 트래픽의 최소 성능 속도를 보장하는 PHB이다. 이러한 AF PHB는 네 개의 클래스로 분류되고, 각 AF 클래스는 어떤 양의 전달 자원(버퍼 공간과 대역폭)이 할당된다.

AF의 트래픽 조절 행동은 트래픽 쉐이핑(traffic shaping), 패킷 탈락(packet dropping), 패킷의 탈락 우선권 증가(low에서 high로)를 포함한다. AF 클래스 내의 IP 패킷은 3가지 가능한 탈락 우선권 값 중 하나로 마킹된다. 즉, 각 클래스는 다시 패킷 탈락 우선 순위에 따라서 세 개의 레벨로 세분되며, 모두 12가지의 서로 다른 패킷 전달 품질을 가지는 PHB로 구성된다.

예를 들면, 평균속도와 최대속도로 정의되는 특정 플로우의 트래픽이 특정 AF 클래스로 서비스 받기로 결정되었다면, 평균 속도 이하로 입력되는 트래픽에 대해서는 가장 낮은 탈락 확률을 적용하고, 평균과 최대 속도 사이의 속도로 입력되는 트래픽에 대해서는 중간의 탈락 확률을 적용하여 서비스를 해줄 수 있다. 이렇게 하나의 특정 AF 클래스 내에는 서로 다른 탈락 확률이 적용되는 세 집합으로 나뉜다.

AF 구현은 패킷을 큐잉 함으로서 단기간 폭주를 처리하지만, 패킷을 탈락시킴으로서 각 클래스 내의 장기간 폭주(패킷 버스트)를 발견하고 대응해야한다. 폭주가 발생한 DS 노드는 높은 탈락 우선권 값을 가진 패킷을 완전히 탈락시킴으로서 더 낮은 탈락 우선권 값을 가진 패킷을 손실로부터 보호하려고 한다. 그래서 DS 노드에서, IP 패킷 전달 보장 레벨은 다음과 같은 사실에 의존한다:

① 전달 자원이 AF 클래스에 얼마만큼 많이 할당되는가?

② AF 클래스와 클래스 내 폭주의 경우 현재 부하가 얼마인가?

③ 패킷의 탈락 우선권이 무엇인가?

AF PHB를 기초로 Gold, Silver, Bronze 서비스로 구성되는 올림픽 서비스의 실현이 가능하다[15]. 적합한 어플리케이션 예로는 운행 업무와 데이터베이스 질의와 같은 시간에 민감한 어플리케이션이 있다. 표준 코드포인트는 그림 8과 같다.

	Class 1	Class 2	Class 3	Class 4
Low Drop Prec	AF11 : 001010	AF21 : 010010	AF31 : 011010	AF41 : 100010
Medium Drop Prec	AF12 : 001100	AF22 : 010100	AF32 : 011100	AF42 : 100100
High Drop Prec	AF13 : 001110	AF23 : 010110	AF33 : 011110	AF43 : 100110

그림 8. AF 코드 포인트

Ⅳ. 스케줄링 알고리즘

통신망의 링크는 수많은 플로우로 구성되어 있으며 각 순간마다 이용 가능한 대역폭은 자원 할당에 의해서 큐에 있는 패킷들간에 공유된다. 공유된 링크 상에서는 여러 플로우들은 각 서비스에 따라 서로 다른 QoS를 요구하기 때문에 각 플로우의 서비스 특징에 따라 적절한 스케줄링이 필요하게 된다. 스케줄링이란 서비스 요구에 따라 구별된 패킷 흐름들을 서비스 시에 개개의 서비스 품질을 만족할 수 있도록 서비스 순서를 결정하는 방법을 말한다[16]. 이러한 종단간 지연과 각 서비스의 QoS를 보장하기 위해서 네트워크 각 노드에서 각각의 서비스에 따라 가중치를 둔 WFQ를 사용할 수 있다.

1. 라우터 구조

에지 라우터에서, Queue Manager는 각각의 패킷을 분류하여 해당 큐에 넣어 주며 어떠한 패킷

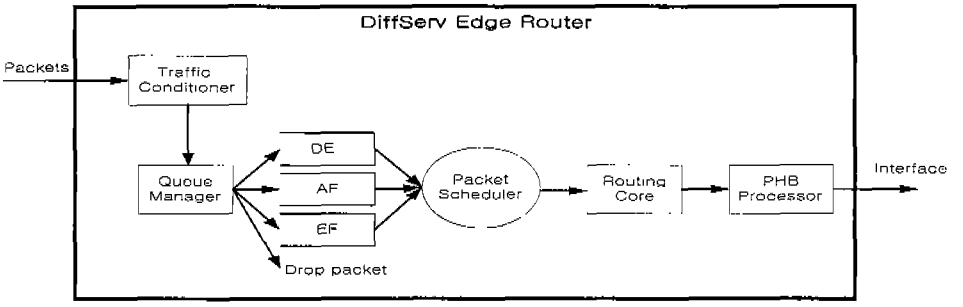

그림 9. 에지 라우터의 개념적 구조

이 요구한 서비스 율보다 많은 도착 율을 가지고 도착한 경우에 shaper/dropper에 의해 탈락시키게 된다(그림 9 참조). 버퍼관리에서 폭주 시 패킷 탈락 방법으로 RED를 주로 사용한다. 코어 라우터에서 RED로 폭주가 되기 전에 임의로 패킷을 제거하고, RED에 의한 패킷 탈락 시에는 큐 길이와 패킷의 계약 준수 여부, 우선 순위들을 고려한다.

TC를 거친 트래픽은 Queue Manager에서 패킷을 분류하여 해당 큐에 넣어주며 패킷 스케줄러가 각 서비스가 가지는 가중치를 얻어서 WFQ를 수행할 수 있다. 라우팅 코어를 거치면 다음 홉이 결정되고 라우팅 코어에서의 처리를 마친 트래픽은 출력 인터페이스를 통해 다음 홉으로 전달되기 전에 차등 서비스 망에서 규정된 패킷에 대한 처리행동 방식인 PHB에 따라 처리를 받게 된다. PHB에 따른 패킷 처리를 담당하는 부분이 바로 PHB 프로세서 모듈이다. 이 PHB 프로세서 모듈을 통과한 트래픽은 비로소 출력 인터페이스를 통해 차등 서비스 망의 다음 홉으로 전달된다.

코어 라우터에서 입력 인터페이스를 통해 입력된 패킷은 라우팅 코어에 의해 출력 인터페이스가 결정되고, 출력 인터페이스가 결정된 패킷들은 해당 출력부에 위치한 PHB 프로세서로 보내진다. PHB 프로세서는 패킷의 DSCP 값을 보고 그 패킷의 PHB를 결정하고 결정된 PHB에 해당하는 패킷 처리를 한 다음 출력 인터페이스를 통해 다음 홉으로 전달한다. 코어 라우터의 동작은 에지 라우터에 비해 상대적으로 간단한 구조를 이루고 있으며 동작에 있어서도 역시 에지 라우터에 비해 간단하다. 일반적으로 망의 경계에서 보다 망의 코어 부분에서 더 많은 패킷이 입출력된다는 점을 고려할 때, 차등 서비스 망의 코어 라우터는 패킷 처리에 있어서 높은 성능을 발휘할 수 있는 기능을 갖추어야 한다(그림 10 참조).

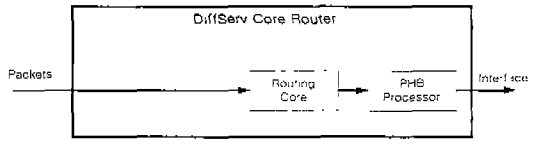

그림 10. 코어 라우터의 개념적 구조

2. 패킷 스케줄링 알고리즘

패킷 스케줄링 알고리즘은 실시간 트래픽의 성능에 중요한 영향을 미치며, 각 플로우에 할당된 만큼의 대역폭을 다른 트래픽의 영향을 받지 않게 보장해 주는 독립성과 모든 플로우가 공평하게 여분의 대역폭을 공유할 수 있는 공평성을 제공해야 한다. 이들은 크게 우선 순위 제어 방식과 대역폭 보장형 스케줄링 방식으로 구분할 수 있다[16]. 스케줄링 알고리즘의 성능은 패킷 대기지연의 상한 값, 공평성 및 구현의 용이성에 의하여 평가할 수 있다. 먼저 우선 순위 제어 방식에 속하는 스케줄링 알고리즘으로는 다음과 같은 것이 있다[16].

- HOL(Head of Line) 기법: 서비스 클래스

별로 버퍼를 설정하고 높은 우선 순위 패킷을 모두 처리하고 해당 클래스의 패킷을 처리하는 방법이다. 낮은 우선 순위 클래스에 대하여 지연 시간이 커지는 단점이 있다.

- WRR(Weighted Round Robin) 기법: 일정 주기 동안 각 입력에 대하여 고정된 가중치를 두어 차별적으로 서비스를 제공하는 방식이다. 가변적인 특성을 가지는 가변 비트 율 서비스를 지원하기에 부적합하다.
- MLT(Minimum Laxity Time) 기법: 버퍼에 대기하고 있는 실시간 패킷의 MLT값이 임계치보다 작을 경우 실시간 트래픽을 처리하고, 그렇지 않은 경우 비실시간 트래픽을 처리하는 방식이다. MLT값의 검색으로 복잡도가 증가된다.
- HOL-PJ(Head of Line-Priority Jump) 기법: 패킷의 해당 버퍼 소비시간이 지정된 시간을 초과할 때 한 단계 높은 우선 순위 버퍼로 점프하여 서비스 받는 것이다.

일반적으로, 우선 순위 제어 방식은 전송 지연의 한계와 대역폭 할당을 독립적으로 설정할 수 있는 장점은 있으나, 우선 순위 적용을 위한 정렬 과정이 복잡하고 다른 플로우에 의해 서로 영향을 받기 때문에 같은 우선 순위를 가진 버스티 트래픽이 존재할 경우 서비스 품질을 보장할 수 없는 경우가 발생한다. 반면에, 대역 보장형 스케줄링 기법은 각 사용자 플로우마다 서비스 품질을 보장하여 주기 위하여 사전에 지정된 비율에 따라 서비스를 받게 하는 공평한 큐잉 기법이라고 할 수 있다.

본 고에서는 몇 가지의 스케줄링 알고리즘에 대하여 DiffServ를 중심으로 보다 자세히 기술하고자 한다.

2.1. Round Robin

RR 스케줄링 알고리즘은 라운드 로빈 방식으로 차례로 플로우별로 서비스를 하는 일종의 Fair Queueing 방법이다. 어떤 버퍼에 패킷이 없을 때는, 다음 버퍼가 검사되고 존재하는 패킷은 스케줄된다. 이런 알고리즘은 DiffServ 구조에 적합하지 못하다. 예를 들면, 두 개의 트래픽 클래스, EF와 BE 서비스가 있다고 하자. EF와 BE 패킷이 큐 안에 있을 때, 패킷은 차례로 디큐(dequeue)되고 긴급하게 디큐되는 것은 없다. 이런 경우 DiffServ 구조의 의도가 아닌 50% EF와 50% BE의 대역폭 공유가 준수된다.

2.2. Priority Round Robin

PRR은 EF 패킷을 가장 높은 우선권을 가지고 디큐한다. 그 다음으로 우선권이 높은 것이 AF 서비스, 마지막으로 BE 패킷이다. 이 알고리즘은 EF가 존재할 때, AF와 BE 트래픽은 아마도 기아 상태가 될 수 있는 단점이 있다. 이는 BE 트래픽이 기아 상태가 되지 않도록(AF 트래픽도 마찬가지) 요구하는 DiffServ의 요구사항에 일치하지 않는 것이다. EF 소스가 너무 많은 트래픽을 전송할 때 EF 트래픽이 100%의 대역폭을 얻을 수 있게 된다. 대역폭은 이런 행동을 방지하는 SLA에 의해 관리되어야 한다.

2.3. Weighted Round Robin

WRR 스케줄링 알고리즘은 RR 방법으로 패킷을 디큐한다. 그러나 패킷 공유에 따라 동일한 큐로부터 차례로 한 개 이상의 패킷을 제거한다. 예로, WRR 알고리즘은 5 EF 패킷, 3 AF 패킷, 마지막으로 2 BE 패킷을 디큐한다. 그리고 다시 5개의 AF 패킷 등으로 시작한다. 이것은 50% EF, 30% AF, 20% BE의 대역폭 공유와 해당한다. 이것은 모든 패킷이 동일한 크기를 가지거나 평균 패킷 크기가 미리 알려졌을 때 동작한다. 그러면 공유 파라미터(여기서는 5, 3, 2)가 적절히 설정된다. 하나의 해결책은 트래픽의 평균 패킷 크기를 계산하는 것이고, 거기에 따라 공유 파라미터를 조정하는 것이다.

최적 공유 파라미터를 발견하는 것은 상당히 복잡하다.

2.4. Weighted Fair Queueing

WFQ 알고리즘은 대역폭 공유 값을 실제 이용되는 대역폭 공유와 비교한다. 그 다음 가장 큰 양(positive)의 차이를 가지는 큐로부터 패킷이 디큐된다. 이런 큐가 큐에 저장된 패킷이 없을 때(트래픽이 존재하지 않을 때), 두 번째로 큰 큐가 디큐된다. 이런 알고리즘으로 받아야 할 평균이상의 대역폭보다 더 서비스를 얻지 못하도록 하는 것을 보장한다. 충분한 가용 대역이 있고 다른 서비스가 없을 때, 존재하는 서비스는 모든 대역폭을 사용할 수 있다. 그러나 다른 서비스가 있을 때는 초과 대역폭은 서비스들 중에서 공평하게 공유된다.

x개의 마지막 패킷을 저장하는 패킷 윈도우의 도움으로, 그들 트래픽 클래스 멤버쉽과 패킷 크기가 정확한 대역폭 공유를 이룰 수 있다. 패킷 윈도우의 크기는 아주 중요한 파라미터이다. 이유는 패킷 윈도우가 작게 선택되었을 때 디큐잉 알고리즘은 RR 디큐잉 알고리즘과 비슷하게 트래픽 클래스들을 충분히 차별화할 수 없기 때문이다. 패킷 윈도우가 너무 크게 선택되었을 때 새롭게 전송되는 패킷은 트래픽 클래스가 똑같이 많은 패킷을 보낼 때까지 전달된다. 그러므로 새롭게 보내지는 패킷은 연속적으로 보내지는 트래픽을 정지시키면서 짧은 시간동안 모든 대역폭을 사용하게 된다. 이것은 특히 실시간 제약을 가지는 EF에게 좋지 않은 것이다. 그러므로 패킷 윈도우는 주의 깊게 선택되어야 한다.

WFQ로 쉐이핑 동작은 실현된다. 하나의 서비스로부터 버스트가 도착할 때 버스트는 큐되지만 버스트로 디큐되지는 않는다. WFQ는 큐잉(TB setting)과 SLA와 매우 잘 일치되어야 한다.

2.5. Priority WFQ

PWFQ는 PRR과 WFQ의 결합이다. EF 트래픽은 엄격한 실시간 제약을 지원하도록 다른 트래픽의 존재여부와 관계없이 가장 높은 우선권으로 디큐된다. 4가지 AF 클래스, BE, 그리고 사실상 네트워크 제어 트래픽은 WFQ 알고리즘으로 디큐된다. EF는 대역폭 공유를 가진다. 그리고 비록 EF가 가장 높은 우선권을 가지더라도 설정된 대역보다는 더 많이 사용할 수 없다. 이 메커니즘은 다른 트래픽이 정규적으로 트래픽을 전송하도록 허용한다. 이 알고리즘은 트래픽 클래스를 우수하게 분리하고 대역폭 공유를 할당할 수 있다.

V. 대역폭 브로커(BB: Bandwidth Broker) 시스템

사용자와 응용에게 고품질 데이터 전달 서비스를 제공하는 것이 통신망 QoS 지원의 궁극적인 목적이다. 그러나 라우터 관점에서, QoS 지원은 세 가지 기본적인 부분으로 구성된다[17]; ① 패킷 처리 클래스 정의, ② 각 클래스를 위한 자원양의 명시, ③ 모든 입력 데이터를 대응하는 클래스로 분류. DiffServ는 위의 첫 번째와 세 번째 문제점을 다룬다. 즉, 트래픽 클래스를 명시하고 간단한 패킷 분류 메커니즘을 제공한다. 라우터는 패킷이 속하는 흐름이나 응용의 유형이 무엇인지 몰라도 ToS 값에 의해 패킷을 대응하는 처리 클래스로 쉽게 분류한다.

BB는 마크된 트래픽의 현재 할당을 추적하고 정책과 현재 할당의 시각에서 새로운 요청을 해석함으로서 두 번째 문제점을 다루기 위한 것이다. BB는 자원관리와 트래픽 제어에 관한 도메인 내부 업무와 외부 관계 모두를 담당할 것이다. 내부적으로, BB는 개별 사용자와 응용으로부터의 QoS 요청을 추적할 수 있다. 그리고 도메인의 특정 자원 사용 정책에 따라 내부 자원을 할당한다. 외부적으로는, BB는 경계를 통과하는 데이터 트래픽의 QoS 취급을 보장하기 위해 이웃 도메인의 BB와 상호 서비스 협약을

설정하고 유지하는 책임이 있다.

1. BB의 기능

서비스 품질을 제공하기 위한 적당한 자원 관리를 위하여, 가용 자원과 입력 SLA를 보고 이런 자원을 어떻게 할당할 것인가를 결정하는 중앙 에이전트가 필요하다. 이런 중앙 에이전트가 BB(Bandwidth Broker)로 알려져 있다. BB는 다양한 SLA와 SLA에 할당된 대역폭을 저장하고, 향후 할당을 결정하기 위하여 기초로 하는 정책 데이터베이스(policy database)를 가진 소프트웨어 개체이다.

BB는 요청된 데로 사용자에게 우선적인 서비스를 할당하고 정의된 서비스를 위하여 올바른 전달 행동으로 통신망 라우터를 구성할 책임이 있는 대리인(agent)이라고 할 수 있다. BB는 도메인마다 하나씩 특정한 신탁 지역(trust region)과 연관된다. BB는 누가 무엇을, 언제 할 수 있고, 요청자를 인증하기 위하여 그 데이터베이스를 사용하는 방법에 관한 정보를 보유하고 있는 정책 데이터베이스를 가지고 있다.

어떤 할당이 특정 흐름을 위해 요구될 때, 요청은 BB에게 보내진다. 요청은 서비스 유형, 타겟 율, 최대 버스트, 그리고 서비스가 요구되는 시간 기간(time period)을 포함한다. 요청은 사용자에 의해 만들어질 수 있고 다른 지역의 BB로부터 올 수도 있다. BB는 먼저 요청자의 자격을 인증하고, 그 다음 요청을 충분히 만족시키기 위한 비 할당 대역폭이 있는지를 검사한다. 만약 요청이 이런 테스트를 통과한다면, 가용 대역폭은 요청된 양만큼 감소되고 흐름 명세가 기록된다.

BB는 서비스가 시작되는 시간에, 서비스가 주어질 패킷 흐름에 관한 정보를 가지고 적절한 리프(leaf) 라우터를 구성한다. 이 배열은 BB가 주기적으로 새로 보급(refresh)할 소프트 상태(soft state)이다.

BB의 개념은 DiffServ 구조의 한 부분으로 소개되었다. BB는 DiffServ 자원을 관리하는 여러 가지 역할을 수행한다. 두 가지 중요한 측면이 있다[19].

- 도메인간(inter-domain) 자원 관리: 도메인간 자원관리는 두 도메인간 네트워크 경계에서 자원을 공급하고 할당하는 것과 관련 있다. 각 측이 송수신을 위하여 동의하는 트래픽의 양과 유형을 명시하는 상호간 SLA가 두 도메인간 경계에서 설정되어야 한다.
- 도메인 내부(intra-domain) 자원 관리: 도메인 내부 자원관리는 통신망이나 도메인 내의 자원 할당을 다룬다.

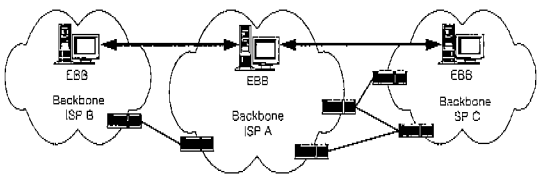

그림 11. 도메인간 BB

통과 네트워크에서, BB는 통로 네트워크의 경계를 통해 도메인으로 들어오고 나가는 DS 트래픽을 추적하고, 인접 도메인과의 상호 동의가 지켜지는 것을 보장한다. BB는 상호 협의에 따라 라우터 내외 트래픽 조절기를 구성하기 위하여 입/출구 경계 라우터와 통신한다(그림 11 참조). 그림 11에서 EBB는 External Bandwidth Broker를 의미한다.

2. BB 시스템의 구조

2.1. BB 시스템의 구성요소

BB 시스템은 다음의 구성요소로 이루어진다[17, 20].

- BB 데이터베이스

- BB 서버
- BB 명령 라인 인터페이스(Command Line Interface: CLI)
  - SLA 클라이언트
  - 자원할당 요청(RAR: Resource Allocation Request) 클라이언트
- BB 라우터 구성 클라이언트

그림 12는 이와 같은 구성 요소를 가진 BB 시스템을 보여주고 있다.

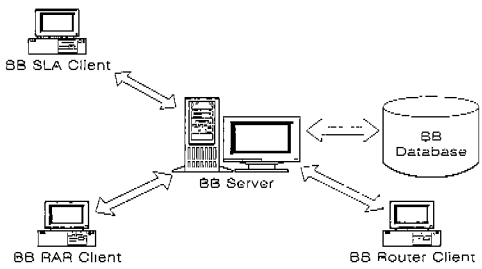

그림 12. BB 시스템의 구성요소

DiffServ 단일 네트워크에서 서버를 구성하는 BB와 여기에 연결되는 DB가 그림 13에서 보여주고 있다. 또한 BB는 클라이언트 인터페이스와 라우터 구성 인터페이스를 가진다.

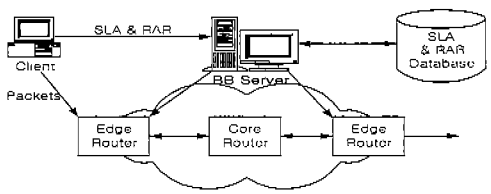

그림 13. BB 인터페이스 개요

2.2. 데이터베이스

이 데이터베이스는 신탁 지역 내 BB의 기능과 관련 있는 모든 데이터를 저장하기 위한 메커니즘을 포함한다. 여기에는 다음과 같은 것이 포함된다:

- Service Level Agreement(SLA)
- Resource Allocation Request(RAR)
- 대역폭 할당과 코드포인트의 매핑

입력은 명령 라인 인터페이스로부터 혹은 클라이언트로부터 요청이 수락될 때마다 데이터베이스에 만들어진다.

2.3. BB 서버

BB 서버는 BB DB와 다중의 동시 BB CLI 클라이언트 사이의 상호 작용을 취급하는 다중 프로세스 프로그램이다. BB 서버는 다음과 같은 기능을 수행한다:

- 클라이언트로부터 어떤 요청(SLA, RAR)이 도착할 때 데이터 검정 및 정책 실행을 수행.
- 만약 요청이 타당하고 할당된 제한 범위 이내라고 나타나면, 대응되는 변경(추가나 갱신)은 DB안의 데이터에 생성.
- BB 클라이언트와 BB DB간 인터페이스를 제공
- 새로운 SLA가 수락될 때마다 클래스를 설정하거나 할당하기 위하여 출구 라우터로 구성 정보를 제공
- RAR이 수락될 때마다 리프 라우터로 구성 정보를 제공. 그것은 수락된 플로우의 폴리싱과 마킹을 실행하기 위한 것이다.

2.4. BB 명령 라인 인터페이스 클라이언트

BB CLI는 BB 서버 및 DB와 상호 작용하기 위하여, 시스템 운영자와 클라이언트를 위하여 다수의 CLI 프로그램을 제공한다. 명령은 다음을 유지하기 위하여 이용 가능하다.

- SLA(Service Level Agreements)
- RAR(Resource Allocation Requests)

1) SLA는 특정 고객에게 특별한 서비스 블록을 할당하기 위한 단순한 메커니즘을 제공한다. 조직이

나 사용자를 위한 장기간 대역폭 예약을 쉽게 한다. 그러나 특정 플로우에 대한 할당을 제한하지 않는다. SLA는 단일 신탁 지역 내의 플로우에 대해 유효하다. 일단 SLA가 설정되면, 부분적인 서비스가 특정 플로우에 할당될 수 있다. SLA는 다음 정보를 포함한다:

- 고객 신원 확인(Customer Identification)
- 서비스 유형(Service Type)
- 서비스 유형 파라미터(Service Type Parameters)
- 서비스 제한(Service Restrictions)

2) RAR은 각 플로우를 위해, SLA에 의해 클라이언트에게 할당된 부분적인 서비스를 요청하기 위하여 사용된다. 이런 개별적인 플로우는 율 정보에 따라 근원지/목적지/프로토콜 정보에 의해 설명된다. RAR은 다음과 같은 기능을 포함한다:

- 사용자 식별
- 협상된 SLA를 위한 SLA ID
- 서비스 레벨 파라미터(율, 최대 버스트, 등)
- 근원지 식별자(포트 번호, IP 주소, 프로토콜)
- 목적지(포트 번호, IP 주소, 프로토콜)
- 요청 기간

요청을 승인하기 전에 BB는 SLA의 할당이 제한을 위반하지 않고, 신탁 지역(trust region)에서 그 서비스의 집약된 양이 초과되지 않도록 보장한다.

2.5. BB 라우터 구성

BB는 DS 지역 내의 요구되는 서비스 레벨을 제공하기 위해 DiffServ가 가능한 라우터 그룹을 구성해야 한다. BB는 서비스 레벨 동의와 클라이언트로부터 입력되는 대역폭 할당 요청에 따라 도메인에 있는 경계 및 출구 라우터를 구성한다. SLA의 확인과 검증 후에, 적합한 출구 라우터는 연결되고 특별한 서비스를 구성하기 위해 요청된 파라미터가 송신된다. RAR의 검증도 비슷하게, RAR에서 언급된 경계 라우터가 연결되고 플로우를 마킹하고 폴리싱하기 위해 필요한 파라미터가 라우터로 전달된다. 그래서 라우터는 서비스 할당이 BB에 의해 변경될 때마다 refresh되어야 한다.

3. DiffServ에서의 BB

3.1. DiffServ 네트워크에서 BB의 역할

BB는 DiffServ 네트워크에 있는 IP QoS 서비스를 위해 자원을 관리하는 소프트웨어 개체이다. BB는 정책 DB에 따라 내부/외부 수락제어를 결정할 책임이 있다. 그런 결정을 기초로, BB는 도메인 내의 라우터를 구성하고 이웃 도메인의 BB와 협상할 책임이 또한 있다.

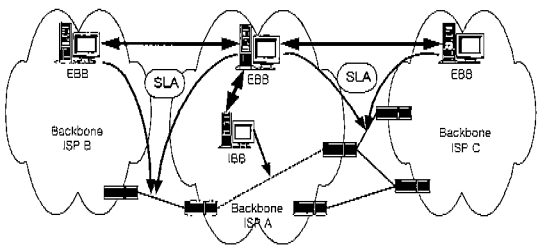

그림 14. BB를 가진 DiffServ 도메인

그림 14는 BB를 가진 DiffServ 도메인을 보여준다. 이런 구현은 SLA/RAR을 위한 요청을 생성하는 클라이언트를 가지는 클라이언트-서버 구조의 형태이다. 시작에서 BB 서버는 도메인에 있는 라우터와 연결될 것이고, 기본 구성을 설정한다. 그리고 도메인 내부 호스트 혹은 네트워크 관리자로부터 입력 요청을 기다린다.

3.2. 기본적인 기능

고객 요청을 기초로 SLA는 일반적으로 네트워크 운영자에 의해 GUI를 통해 BB에 추가된다. 고객은 도메인 내의 호스트이거나 이웃 도메인의 네트워크 운영자이다. 그래서 BB는 SLA 설정에 직접 포함되지 않는다. 네트워크 운영자는 GUI를 사용해 DB안의 입력을 보거나 추가, 수정, 삭제할 수 있다. BB는 SLA를 수락하기 전에 제한(limit) 체크를 수행한다. 만약 SLA가 허락될 수 없다면 단지 그것을 거절하는 대신에 네트워크에서 현재 가용 대역폭을 반환한다.

RAR은 DiffServ를 요구한 어플리케이션을 수행하는 호스트에서 생성된다. BB는 요청이 제한을 초과하지 않도록 하는 것을 보장하기 위하여 같은 SLA에 대하여 대응되는 SLA과 다른 RAR에 대해 비교한다. 만약 제한 내이라면, BB는 필요하면 에지 라우터 구성을 수행하고 플로우에 설정될 DSCP와 함께 호스트로 성공 메시지를 반환한다. 플로우는 진행되고, 그 플로우의 패킷은 코드포인트로 마킹된다. 실패하는 경우에는, 플로우가 거절된다.

BB는 또한 플로우와 협약에 대한 유효한 기간을 추적한다. 플로우가 만료일 때, 플로우를 위해 할당된 자원은 해제된다. SLA가 만료일 때, SLA를 위한 모든 RAR 목록들은 SLA과 함께 제거된다. 또한 다른 SLA, RAR에 관한 정보, 도메인에 관한 다른 정보를 저장하는데 사용하는 DB를 유지한다.

코어 라우터는 패킷 분류와 전달을 위해 정적으로 구성된다. 에지 라우터만이 플로우 개시 시에 BB에 의해 폴리싱과 쉐이핑을 위해 구성될 것이다.

3.3. 사용자/클라이언트-BB 상호작용

각 호스트에서 수행되는 DS daemon은 호스트에 있는 어플리케이션으로부터 QoS 요청을 다룬다. 어플리케이션은 대역폭 요구사항을 결정해야 한다. 어플리케이션 요청 내의 파라미터는 다음과 같다: SLA_ID, 서비스 유형, 율 파라미터, 목적지 주소, 목적지 포트, 프로토콜, 흐름 기간.

3.4. BB-라우터 상호작용

BB는 자신의 DB에 도메인에 있는 라우터에 관한 정보를 가진다. 내부 링크에 적합한 용량이 있고 BB 내에 개별적인 링크 용량과 라우팅 토폴로지 정보를 유지할 필요가 없다고 가정하면, 단지 에지 라우터만 DB에 포함될 필요가 있다. 내부 라우터는 통용되는 정책을 기초로 정적으로 구성된다.

BB에 의해 유지되는 정보는 라우터의 기본 IP 주소, 라우터 유형, IP 주소에 따라 다른 인터페이스 이름을 포함한다. BB는 라우터와 연결하기 위해 기본 IP 주소를 사용한다. 나머지 정보는 라우터 구성을 위해 사용된다. 그 정보는 라우터 자신으로부터 직접 얻을 수 있다. 그러나 DB에 수동으로 채워진다.

RAR로부터 목적지 주소가 주어지면, BB는 에지 라우터에 있는 대응되는 라우팅 테이블 입력(entry)을 얻는다. 그리고 DB에 있는 ROUTE 테이블을 갱신한다. 이 입력은 RAR의 종료에 따라 제거된다.

Ⅵ. DiffServ와 ATM

1. ATM 서비스의 범주

ATM은 이종 QoS 레벨을 지원하는 5가지 서비스 범주를 가진다. CBR과 Rt-VBR은 실시간 서비스를 지원하기 위해 설계되었으며, Nrt-VBR, UBR, ABR은 비실시간 서비스에 적합하다.

(1) CBR(Constant Bit Rate) 트래픽

CBR은 일정하거나 혹은 최대의 비트율로 실시간 데이터를 전송하기 위하여 설계되었다. PCR(Peak Cell Rate)로 트래픽을 간단히 묘사할 수

있다. 그러므로 일정한 양의 고정된 대역폭을 할당하여야 한다. PCR과 평균 비트율이 동일하므로 CBR 트래픽의 버스트성은 1이 된다. CBR 트래픽은 엄격한 지연 및 지연 변화에 대한 요구조건을 가지며, 실시간 어플리케이션에 적용된다. 최대 셀 전달 지연을 초과하여 전달되는 셀들은 실시간 정보로서의 의미를 잃게 된다. 트래픽 마킹, 폴리싱, 쉐이핑은 실시간 트래픽 지원을 위해 중요하다.

(2) Rt-VBR(Real-time Variable Bit Rate) 트래픽

Rt-VBR은 버스티 실시간 링크를 위해 만들어졌다. 시간에 민감한 특성을 가지며, 그 예로 음성 및 비디오 어플리케이션 등이 있다. Rt-VBR은 실시간 소스들의 통계적 다중화를 지원할 수 있고 지속적으로 보장되는 QoS를 제공할 수도 있다. MBS(Maximal Burst Size)는 다른 모든 CBR 파라미터 이외에 추가로 명시된다. 추가적인 파라미터의 명세는 허용되는 최대 버스트를 제한한다; 그러므로 큐 공간 예약은 버스트를 보유하도록 만들어진다.

(3) Nrt-VBR(Non-real-time Variable Bit Rate) 트래픽

Nrt-VBR은 셀이 우선권을 가지고 전달되는 버스티 비 실시간 트래픽을 위하여 유용하다. 그리하여 critical response time을 가지는 링크가 이런 서비스를 사용할 수 있다. 트래픽 계약 범위 내에서 전송되는 셀들은 낮은 손실률을 가진다. 모든 셀들에 대해서 셀 전달 지연 경계치(bound)가 있으며 연결들의 통계적 다중화를 지원한다. Rt-VBR을 위한 것과 같은 동일한 트래픽 파라미터가 설정되어야 한다. CLR(Cell Loss Ratio)은 송신자가 동의된 파라미터를 초과하지 않는다면 보장된다. CLR은 Nrt-VBR을 위해 명시된 유일한 QoS 파라미터이다.

(4) UBR(Unspecified Bit Rate) 트래픽

UBR은 FTP와 같은 전통적인 컴퓨터 통신 어플리케이션을 위해 만들어졌다. 그래서 엄격한 지연 및 지연 변화에 대한 요구사항이 없는 지연 허용 혹은 비실시간 어플리케이션을 위한 트래픽이다. CLR(Cell Loss Ratio)과 CDT(Cell Transfer Delay)에 관한 어떠한 규약도 만들어지지 않는다; UBR 링크의 공유는 반드시 공평하지는 않고, 특정한 트래픽 계약도 없다. 심지어 데이터 전송에 관한 약속도 전혀 없다. 최소한의 QoS 지원을 가지는 트래픽이고 인터넷상의 전통적인 Best-Effort 트래픽과 비교될 수 있다. 파라미터 PCR과 CDVT(Cell Delay Variation Tolerance)는 명시되지만 QoS 협의는 없다.

(5) ABR(Available Bit Rate) 트래픽

ABR은 UBR과 같은 동일한 트래픽을 전송하지만 흐름 제어를 통해 낮은 폭주 확률을 가진다. 통신망 상의 가용 비트율에 따라서 정보 전달을 조절할 수 있는 어플리케이션을 위한 트래픽에 해당한다. 흐름 제어는 링크가 네트워크에서 가용 비트율에 따라서 비트율을 조정할 수 있는 메커니즘이다. 시간적으로 가용 대역폭에 적응할 수 있는 트래픽 형태이고, 지연의 변화에는 둔감하나 손실에는 아주 민감한 트래픽이다. 낮은 CLR은 트래픽 계약 내에 머무를 수 있는 스테이션에 대하여 기대할 수 있고 수신자로부터 피드백을 통해 수행되는 흐름 제어를 가진다. 가용 대역폭은 MCR(Minimum Cell Rate)에서 PCR까지 다양하다. ABR 서비스는 MCR을 보장하면서 CBR 및 VBR 트래픽들이 사용하지 않는 링크의 가용 대역폭을 최대한 활용하기 위한 서비스이다. 트래픽 계약은 양방향에서 협상되고 네트워크는 공평한 자원 공유를 약속한다. 이런 사전 조건을 가지고 링크는 폭주 상태의 ATM 네트워크에서도 규칙적인 서비스를 기대할 수 있다.

2. DiffServ와 ATM 서비스의 매핑

가장 좋은 매핑을 개략적으로 기술하면 다음과 같다[22-27].

(1) Best-Effort와 UBR/ABR

Best-Effort는 UBR과 ABR 모두에 어울린다. 그러나 응용이나 하부 링크 레벨이 폭주를 피하기 위하여 흐름 제어를 지원하는 한 ABR이 항상 더 바람직하다. FTP, HTTP, Telnet, E-mail 및 news가 이런 서비스 조합을 가진 어플리케이션에 적합하다.

(2) 프리미엄 서비스와 CBR

CBR을 가지는 프리미엄 서비스는 가상 전용선(VLL: Virtual Leased Line)을 형성한다. 그것은 오디오와 비디오 전송과 같은 일정한(constant) 흐름에 적합하다. 전체적인 평균 비트율이 아닌 MPEG과 같은 압축 데이터를 가진 실시간 어플리케이션도 지원될 수 있다. 이런 링크 상으로, 상대적으로 큰 버퍼를 가진 에지 라우터에 있는 폴리싱과 쉐이핑은 비트율을 일정하게 만들기 위해 첨두(peak)를 평활화 시킨다.

(3) 보장형 서비스와 nrt-VBR

보장형 서비스와 nrt-VBR은 버스티 비실시간 트래픽을 지원하는 서비스 조합이다. 이런 서비스는 적합한 링크 행동을 은행 거래, 항공 예약시스템, 가상 현실, 원격 수술, HTTP, Telnet과 같은 응답 시간이 아주 주요한 어플리케이션으로 가져다준다.

(4) DiffServ서비스, PHB Group과 ATM 서비스간의 매핑

표 2는 위에서 설명한 최상의 매핑을 기초로 한 매핑을 보여준다.

표 2. 차별서비스와 ATM 서비스간 최적의 매핑[22]

DiffServ 서비스	ATM
BE 서비스	UBR / ABR
프리미엄 서비스	CBR
보장형 서비스	Nrt-VBR
올림픽 서비스	특별한 매핑이 없음
SRP 서비스	CBR
USD 서비스	Nrt-VBR / ABR

표 3은 표준 PHB Group과 ATM 서비스의 가장 좋은 매핑을 보여준다.

표 3. PHB Group과 ATM 서비스간 최적의 매핑[28]

PHB Group	ATM
DE	UBR / ABR
AF	Nrt-VBR
EF	CBR

3. DiffServ over ATM 구현

3.1. AF over ATM

정규 AF 서비스를 위하여, FH(First Hop) 라우터는 해당하는 AF 클래스와 함께 어떤 프로파일에 따라 패킷을 분류한다. 탈락 우선권은 흐름이 low, medium, high 탈락 우선권에 대한 어떤 비트율 제한을 초과하는지에 의존하여 선택된다. AF over ATM의 매핑에 대해 두 가지 옵션이 존재한다[30]: 첫 번째 옵션은 모든 마킹 기능을 그대로 가진 AF 라우터 구현을 유지한다. AF 트래픽은 다음 홉까지 어떤 AF PVC를 통해 전달된다. 다시 전달 행위는 IP 목적지 주소이외에 IP 소스 주소, DSCP 등과 같은 정보를 고려해야 한다. 탈락 우선권이 낮은 패킷은 PVC의 CLP=0인 셀로 매핑되고, 탈락 우선권이 medium과 high 패킷은 CLP=1인 셀로 매핑된다.

두 번째 선택은 보장형 서비스 패킷의 올바른 마킹을 위해 ATM 스위치에서 구현된 리키 버킷 메커

니즘 사용을 시도한다. 두 번째 접근은 두 가지 옵션으로 가능하다:

① 입출력 라우터 사이에서 ATM 스위치 내부의 쉐이핑과 폴리싱 기능이 사용될 수 있다. 즉, CLP=1로서 비준수(non-conforming) 셀을 마킹한다. IR(Ingress Router)에서, CLP=0을 가진 셀로 구성된 패킷은 다른 패킷이 높은 탈락 우선권 값을 얻는 반면에 낮은 탈락 우선권 값을 가져야 한다. 이것은 low와 high 탈락 우선권만이 사용되고 medium 탈락 우선권은 사용되지 않는 단점을 가진다. 첨두 비트율을 약간 초과하는 경우, 각 패킷은 CLP=1로 마킹되어진 한 개의 셀을 항상 가질 수 있다. 이것은 더 많은, 심지어 모든 패킷이 정규 DiffServ 경우보다 높은 탈락 우선권을 가지도록 하는 마킹을 초래할 수 있다.

② 다른 접근은 CLP=1인 셀의 수가 어떤 임계치 아래일 때 낮은 탈락 우선권으로 입구 라우터에서 패킷을 마크하고, 반면에 CLP=1인 셀의 수가 다른 임계치를 초과할 경우에 패킷은 높은 탈락 우선권으로 마크된다. 다른 모든 패킷은 중간 탈락 우선권으로 마크된다. 이러한 스킴은 IP 라우터에서 패킷 기반 위에서 수행되는 마킹 스킴과 일치하도록 만들어져야 한다. 또한 CLP 마킹을 기초로 한 폴리싱 기능이 ATM 교환기 내에서 이용 가능한 것을 요구한다.

3.2. EF over ATM

EF 서비스인 경우, 고객 네트워크에 있는 소위 FH(First-Hop) 라우터는 EF 흐름을 분류, 적합한 DSCP로 EF 흐름을 마킹하고, 고객과 ISP 사이에 사전에 협상된 EF 첨두율에 따라 흐름을 쉐이핑해야 한다[30]. 에지 라우터는 전체 EF 트래픽을 미리 협약된 율에 따라 쉐이프해야 한다. 그 다음 ISP에 있는 내부 라우터는 고객과 사전 협상된 프로파일을 기초로 폴리싱을 수행한다.

한 가지 접근은, FH는 FH 내에 구성된 정의된 프로파일 정보에 따라 모든 EF 패킷을 분류하는 것이다. 다음 단계에서 다양한 흐름은 준수된 흐름이 되기 위해 쉐이프 되어야 한다. FH에서 쉐이핑을 위해, FH와 에지 라우터간에 PVC를 설정한다. 각 PVC에 대해, 양쪽 라우터에서 Classical IP over ATM 인터페이스를 생성한다. 이런 각각의 논리적인 ATM 인터페이스 주소 쌍은 공통의 IP 서브넷을 공유한다. IP over ATM 인터페이스 쉐이핑 율로 구성될 수 있다.

3.3. DiffServ over ATM을 위한 모델

DiffServ over ATM 모델을 위해서는, 입력되는 패킷은 먼저 분류되고 어떤 출력 인터페이스/디바이스로 전달된다. 이런 디바이스를 위한 DiffServ 큐잉 구성요소는 패킷이 얻어야 할 서비스에 따라 패킷을 마킹하기 위해서 선택된다. 라우터 구현의 첫 부분에서 패킷을 전달한 후에, 패킷은 적당하게 마킹되고, 라우터 구현 두 번째 부분에 있는 전용 ATM PVC를 통해 전송된다.

라우터 구성이외에, 두 라우터간 ATM 스위치는 적절히 구성되어져야 한다. 즉, DiffServ 트래픽을 위해 사용되는 PVC가 설정되어져야 하고, 관련 있는 쉐이핑과 폴리싱 기능이 네트워크 관리 기능에 의해 구성되어져야 한다.

그림 15는 ATM 지원이 없는 현재의 DiffServ 구조 모델이다. 그러한 정규 DiffServ 모델에서, 패킷은 마킹, 쉐이핑, 혹은 폴리싱을 포함하는 서비스 핸들러에 의하여 디바이스로 전달되고, 분류되고, 처리되며, 마지막으로 전송을 위해 큐잉된다. WFQ나 우선 순위 스케줄링 같은 큐잉 메커니즘은 다른 큐로부터 패킷을 수집해야 하고 출력 인터페이스를 통한 전송을 위해 패킷들을 스케줄링 한다.

그림 16은 만약 그림 15의 두 번째 인터페이스가 ATM 인터페이스와 대치된다면 그림 15의 모델과 동일한 구조를 보여준다. 이 경우에, 트래픽은 초기에 분류되고, 어떤 패킷은 특정 논리적 ATM 인터

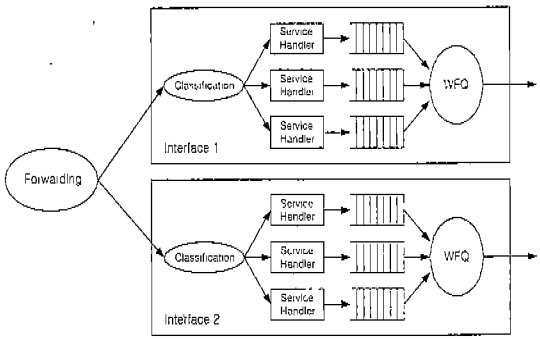

그림 15. ATM 지원 없는 DiffServ 모델

페이스로 직접 전달된다. 단지 하나의 흐름만이 논리적 ATM 인터페이스에 의해 처리되기 때문에, 더 이상의 분류는 요구되지 않는다. 이런 논리적 출력 인터페이스를 통해 전송되는 패킷은 단일 큐로부터 오기 때문에 WFQ와 같은 출력 큐잉은 요구되지 않는다. 이런 인터페이스와 함께 사용되는 서비스 핸들러는 패킷이 적절히 전송되도록 마킹하기 위해 사용될 수 있다.

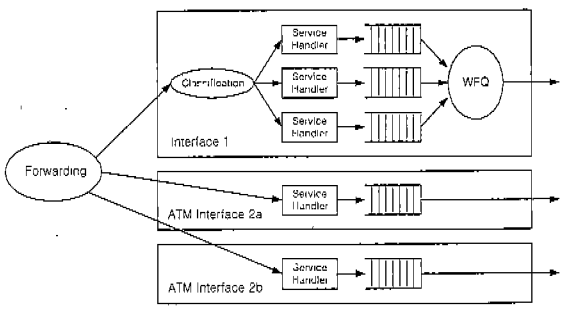

그림 16. DiffServ over ATM 모델

Ⅶ. 결 론

인터넷 QoS 보장을 위하여 통합 서비스 모델이 IETF에서 표준화가 되었다. 그러나 통합 서비스 모델에서는 RSVP라는 신호 프로토콜을 이용하여 패킷 스트림 혹은 플로우별로 자원을 할당하기 때문에 인터넷 백본 망에서와 같이 플로우의 수가 많은 경우에 확장성 문제로 실제로 적용하는데 문제가 발생한다. 이를 해결하기 위하여 QoS 보장을 플로우가 아닌 가입자와 같은 집합(aggregate)으로 분류하여 해주는 DiffServ에 대한 연구 및 표준화가 진행되고 있다.

DiffServ의 특징은 네트워크를 에지와 코어로 분류하여, 복잡한 패킷 분류 및 선처리를 트래픽의 강도가 낮은 에지에서 수행하고, 코어에서는 에지에서 처리되어 패킷 헤더에 삽입된 DSCP에 대응되는 PHB를 적용함으로써 고속의 패킷 전달을 수행한다.

본 논문에서는 DiffServ를 이용하여 QoS를 보장할 수 있는 DiffServ 구조 모델, 홉 별 행동(PHB), 그리고 서비스 요구에 따라 구별된 패킷 흐름들을 서비스해주는 방법인 스케줄링 알고리즘과, SLA을 위한 BB 시스템, 그리고 DiffServ와 ATM 서비스의 매핑에 대하여 기술하였다.

인터넷에서의 서비스 품질 보장 문제는 매우 어려운 문제이다. 그러나, 인터넷을 통하여 인터넷 전화, 인터넷 방송, VPN, 멀티미디어 등의 서비스를 제공하기 위해서는 인터넷이 서비스 품질 보장 기술을 가지는 것이 필수적이다. 그러므로, 품질 보장 기술에 관련된 기술을 국내에 정착시키기 위하여 보다 많은 연구가 필요하다고 하겠다.

※참고문헌

[1] X. Xiao, et al., "Internet QoS : A Big Picture", IEEE Network, pp.8-18, March/April 1999.

[2] W. Weiss, "QoS with Differentiated Services", Bell Labs Technical Journal, Oct-Dec, 1998.

[3] J. Sikora, et al., "Differentiated

Services for Internet2", http://www.internet2.edu/qos/may98 Workshop/html/diffserv.html, 1998.

[4] S. Shenker, et al., "Specification of Guaranteed Quality of Service", RFC 2212, Sep, 1997.

[5] Y. Bernet, et al., "A Conceptual Model for Diffserv Routers", Internet Draft, June, 1999.

[6] K. Nichols, et al., "A Two-bit Differentiated Services Architecture for the Internet", RFC 2638, July, 1999.

[7] J. Heinanen, et al., "An Assured Forwarding PHB Group", RFC 2597, June, 1999.

[8] V. Jacobson, et al., "An Expedited Forwarding PHB", RFC 2598, June, 1999.

[9] S. Blake, et al., "An Architecture for Differentiated Services", RFC 2475, Dec., 1998.

[10] K. Nichols, "Definition of Differentiated Services Behavior Aggregates and Rules for their Specification", Internet Draft, Feb. 2000.

[11] G. Mamais, "Efficient Buffer Management and Scheduling in a Combined IntServ and DiffServ Architecture: A Performance Study", 1999, IEEE.

[12] K. Nichols, et al., "Definition of the Differentiated Services Field(DS Field) in the IPv4 and IPv6 Header s", RFC 2474, Dec., 1998.

[13] Loukola M.V., et al., "Dynamic RT/NRT PHB group", Internet Draft <draft-loukola-dynamic-00.txt>, Nov., 1998.

[14] Nandy B., et al., "Diffserv's Assured Forwarding PHB: What Assurance does the Customer Have?", NOSSDAV' 99.

[15] S. Blake, "Differentiated Services Operational Model and Definitions", draft-nichols-dsopdef-00.txt, Feb. 1998.

[16] 한국전자통신연구원, "실시간 인터넷 서비스 제공을 위한 패킷 스케줄링 연구동향".

[17] D. Sreekantan, et al., "Implementation of a Bandwidth Broker System for Resource Management in Differentiated Services", http://www.ittc.ukans.edu/~kdrao/845, 1999.

[18] "CA*netⅡ Differentiated Services - Bandwidth Broker High Level Design", Apr. 1999.

[19] R. Neil, et al., "A Discussion of Bandwidth Broker Requirements for Internet2 Qbone Deployment", V7. Aug. 1999.

[20] "CA*netⅡ Differentiated Services - Bandwidth Broker System Specification", Oct. 1998.

[21] "CA*netII Differentiated Services - Bandwidth Broker High Level Design", Apr. 1999.

[22] A. Dobreff, "Differentiated Services in ATM Networks", http://www.iam.unibe.ch / ~rsv / publications, Oct. 1998.

[23] W. Almesberger, et al., "Scalable Resource Reservation for the Internet", draft-almesberger-srp-00.txt, Nov. 1997.

[24] Baker, et al., "IP Precedence in Differentiated services Using the Assured Service", draft-ietf-diff-serv-precedence-00.txt, April, 1998.

[25] S. Shenker, et al., "Specification of Guaranteed Quality of Service", RFC 2212, Sep, 1997.

[26] M. Borden, et al., "Interoperation of Controlled-Load Service and Guaranteed Service with ATM", RFC 2381, Aug. 1998.

[27] Z. Wang, "User-Share Differentiation(USD) Scalable bandwidth allocation for differentiated service s", draft-wang-diff-serv-usd-00.txt, Nov. 1997.

[28] A. Dobreff, "Comparison of Simulated and Real Functionality for the Mapping of Differentiated Services to ATM", Nov. 1999.

[29] M. May, et al., "Simple Performance Models of Differentiated Services Schemes for the Internet", IEEE, 1999. pp 1385-1394.

[30] T. Braun, et al., "Implementation of Differentiated Services over ATM", Conference on High Performance Switching & Routing (Joint IEEE ATM Workshop 2000 and 3rd International Conference on ATM), June 26-29, 2000.

전 용 희

1978년 고려대학교 전기공학과 졸업(BS)
1989년 미국 노스캐롤라이나주립대 대학원 Elec. and Comp. Eng. 졸업(MS)
1992년 미국 노스캐롤라이나주립대 대학원 Elec. and Comp. Eng. 졸업(Ph. D.)
1978년~1978년 삼성중공업(주) 근무
1978년~1985년 한국전력기술(주) 근무
1989년~1989년 미국 노스캐롤라이나주립대 Dept of Elec. and Comp. Eng. TA
1989년~1992년 미국 노스캐롤라이나주립대 부설 CCSP(Center For Comm. & Signal Processing) RA
1992년~1994년 한국전자통신연구원 교환전송기술 연구소 선임연구원
1994년~현재 대구가톨릭대학교 공과대학 컴퓨터 · 정보통신공학부 부교수
관심분야 : 차세대 인터넷, 초고속 통신망 프로토콜, 통신망 성능분석, QoS 보장 기술, 고속 통신망 응용 서비스, 통신망 보안

박 수 영

1991년 대구가톨릭대학교 전산통계학과 졸업(학사)
1996년 대구가톨릭대학교 대학원 전산통계학과 졸업(석사)
1999년 대구가톨릭대학교 대학원 전산통계학과 박사과정 수료
1990년~1992년 대구백화점(주) 정보시스템부 근무
1996년~1999년 대구미래대학 멀티미디어정보과학과 겸임교수
관심분야 : 차세대 인터넷, 초고속 통신망 프로토콜, QoS 보장 기술, 통신망 성능분석