

웹 서비스 처리 성능 개선을 위한 웹 서버에서의 우선순위 할당 기법

Priority Allocation Scheme for Improving Web Services Performance on Web Server

김동준(DongJoon Kim)*, 이상규(SangKyu Lee)*, 한상용(SangYong Han)**

초 록

최근에 많은 기업들은 XML 메시지 기반의 분산 환경에서 이기종 간의 표준으로 자리 잡고 있는 웹 서비스를 도입하여 B2C 뿐만 아니라 B2B에 걸쳐서 기업과 기업간 비즈니스를 수행하고 있다. 그래서 웹 서비스의 도입이 성공적으로 이루어지기 위해서는 차별화된 서비스 품질을 보장해 주어야 한다. 하지만, 현재 대부분의 웹 서비스에서는 메시지에 대하여 차별화를 두고 있지 않으며, 기존의 웹 서버들은 웹 서비스 제공자와 사용자간에 체결된 서비스 수준 계약에 대한 품질 요소들을 적절하게 반영하지 못하고 있다.

본 논문에서는 차별화 서비스에 사용되는 응용 수준에서의 적합한 품질 요소를 분석하고, 이러한 품질 요소를 이용하여 웹 서비스 메시지를 처리하는 프로세스에 대하여 우선순위를 할당할 수 있는 기법을 제안한다. 이 기법은 서비스 수준 계약을 최대한으로 만족시키기 위하여 동적으로 우선순위를 할당할 수 있다.

ABSTRACT

The web services concept is rising as a standard to integrate heterogeneous systems in distributed environment based on XML messaging. Recently, for that reason, many companies are trying to use web services for their B2C and B2B business. In order to successfully introduce the web services, it is required to guarantee a differentiated quality of web services. However, most of the current web services cannot provide differentiated policies for requested messages and the existing web servers cannot consider appropriately the quality factors of web services in the service level agreement between provider and consumer.

In this paper, we analyzed appropriate quality factors of web services in application level and suggested priority allocation strategy for processes dealing with messages of web services using those factors. This strategy can allocate dynamically a priority to satisfy the service level agreement.

키워드 : 웹 서비스, 차별화 서비스, 서비스 품질, 서비스 수준 계약(SLA)

Web Services, Differentiated Service, Quality of Service

* 중앙대학교 컴퓨터공학과

** 중앙대학교 컴퓨터공학과 교수

1. 서 론

인터넷은 많은 사람들이 보편적으로 사용하고 있으며, 최근에 많은 기업들은 XML 메시징 기반의 분산 환경에서 이기종 간의 표준으로 자리 잡고 있는 웹 서비스를 도입하여 B2C 뿐만 아니라 B2B에 걸쳐서 기업과 기업 간 동적인 비즈니스를 수행하고 있다. 하지만, 웹 서비스의 도입이 성공적으로 이루어지기 위하여 반드시 필요한 것은 웹 서비스 사용자가 만족할 수 있는 서비스 품질을 보장해야 한다는 것이다. 웹 서비스의 품질을 보장하기 위하여 서비스 공급자는 다양한 서비스 수준에 따라 차별화된 웹 서비스를 제공할 수 있어야 하며, 이러한 다양한 서비스 수준을 보장하기 위해서는 반드시 사용자와 제공자간의 서비스 수준 계약(SLA: Service Level Agreement)이 필요하다[5]. 서비스 수준 계약은 웹 서비스 사용자와 제공자 사이의 책임 관계를 정의하고 제공하는 서비스의 신뢰성을 보장하기 위하여 이들 상호간에 체결하는 것으로, 웹 서비스 제공자는 서비스 수준 계약에 체결된 웹 서비스 품질을 보장할 수 있어야 한다.

하지만, 현재의 웹 서비스 기술 표준 단체들은 웹 서비스 품질을 평가하기 위한 요소 혹은 평가된 정보를 기술하기 위한 목적의 언어에 대하여 표준을 만들어내지 못하고 있으며, 단지 몇몇 매터들이나 학계에 의하여 개별적인 명세화와 연구가 진행되고 있다[2, 3, 4, 6, 7]. 웹 서비스 품질은 성능, 신뢰성, 가용성, 보안 등과 같이 기능적 서비스를 제공할 때의 서비스의 수준을 나타내며, 품질에 대한 분류

기준에 따라서 다양하게 존재할 수 있다.

한편, IETF(Internet Engineering Task Force)에서는 네트워크 수준에서 차별화된 서비스의 품질을 보장하기 위하여 차별화 서비스(DiffServ: Differentiated Service)를 제안하였다[10]. 하지만 차별화 서비스에서는 인터넷을 경유한 종단간(end-to-end) 전송 품질의 보장이 되지 않으며, 통신망의 각 구간별 차별화 패킷 전송 처리 기능만이 구현되었다. 이러한 점을 보완하기 위하여 최근에는 웹 서버에서 차별화 서비스를 제공하려는 연구들이 많이 진행되고 있다[8, 9, 11, 12]. 특히 웹 서비스에서는 SOAP 메시지의 파싱시간과 비즈니스 로직을 실행하는 시간이 존재하기 때문에 네트워크 지연시간보다 웹 서버에서의 지연시간이 커지는 경향이 많아 지고 있으며, 이 때문에 웹 서버에서의 차별화 서비스는 더욱더 중요하다. 이러한 연구들은 네트워크 수준뿐만 아니라 애플리케이션 수준의 서비스를 위하여 사용자의 요청에 따라 프로세스들을 스케줄링하는 방안들을 연구하고 있다. 현재 대부분의 웹 서버들은 FIFO, static priority 등의 스케줄링 방식을 사용하고 있다. 하지만, 이들 방식은 각 상황에 맞게 동적으로 우선순위를 할당하지 못하여 우선순위가 낮은 프로세스에 대해 기근(starvation) 문제를 발생 시키거나 과거에 서비스되었던 품질 정보에 대한 성능 평가가 반영되지 못하고 있다.

본 논문에서는 차별화된 웹 서비스를 애플리케이션 수준에서 제공하기 위하여 다양하게 존재하고 있는 웹 서비스 품질 요소들을 분석하여 차별화 서비스에 사용되는 적합한 웹 서비스 품질 요소를 정의하고, 이러한 품

질 정보를 이용하여 각 상황에 맞게 메시지를 처리하는 프로세스에 대하여 동적으로 우선 순위를 할당함으로써 서비스 수준 계약을 최대한 만족할 수 있는 스케줄링 기법을 제시한다. 웹 서비스가 꼭 웹 서버에서 동작해야 하는 것은 아니지만, 현재 대부분의 웹 서비스는 웹 서버를 이용하여 상호작용을 하고 있다. 이에 본 논문에서도 웹 서버의 환경에서 차별화 서비스를 제공하기 위하여 요청 메시지에 대하여 우선순위를 할당할 수 있는 방법을 제안하고자 한다.

2. 관련 연구

2.1 웹 서비스 품질

웹 서비스의 중요성이 증대됨에 따라 서비스의 품질은 서비스의 사용 빈도와 상호간의 신뢰성에 중요한 영향을 미치게 되며, 이는 서비스 제공자의 성공을 위한 중요한 요소가 되어가고 있다. 현재의 웹 서비스 기술 표준화 단체들은 웹 서비스 품질을 평가하기 위한 요소 혹은 평가된 정보를 기술하기 위한 목적의 언어에 대한 노력이 미약한 실정이며, 단지 몇몇 벤더들에 의하여 개별적인 명세화가 진행되고 있다[3]. 웹 서비스 품질은 성능, 신뢰성, 가용성, 보안 등과 같이 기능적 서비스를 제공할 때의 서비스의 수준을 나타내며, 품질에 대한 분류 기준에 따라서 다양하게 존재할 수 있다.

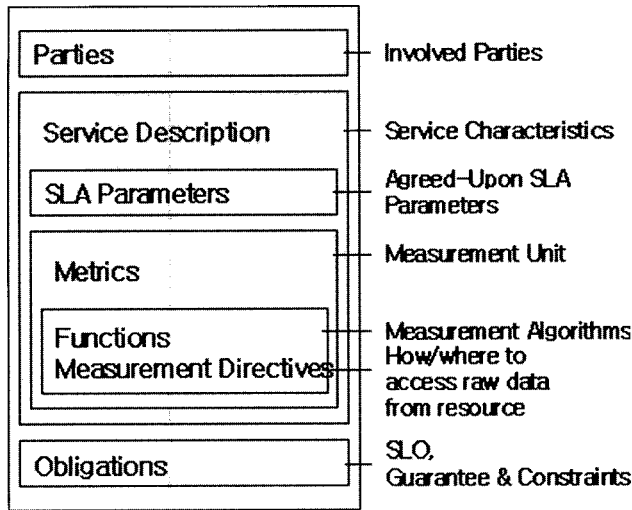
웹 서비스 품질에 관한 표준화 동향으로는 크게 성능 및 안정성 측면과 웹 서비스 및 플

랫폼에 대한 관리 측면의 품질 표준화로 구분할 수 있다[2]. 성능 및 안정성 측면에 관한 표준은 표준화 단체에서 진행되는 것은 아니지만, IBM에서 WSLA(Web Services Level Agreement)[3]로 표준화하고 있으며, 웹 서비스 및 플랫폼에 관한 관리 측면의 품질 표준화는 HP의 WSMF와 IBM의 WS-Mangeability로 각각 표준화되었다가 OASIS의 WSDM(Web Services Distributed Management)으로 통합하였다.

2.2 서비스 수준 계약

서비스 품질 요소들을 기반으로 웹 서비스의 성능을 정의된 수준으로 보증하기 위한 서비스 제공자와 사용자간의 공식적인 계약이 필요하다. 이와 같은 서비스 수준 계약은 포괄적일 수도 있고 매우 상세화될 수도 있다. 어떤 경우 고객은 개별화된 서비스 수준 계약을 통해 서비스 제공자가 보증한 특정 성능 수준을 기대하기도 하며, 계약 이행에 실패한 경우 서비스 제공자와 사용자가 취해야 하는 단계들을 포함할 수 있다[2, 5].

IBM에서는 웹 서비스 환경의 서비스 수준 계약의 표준과 서비스 수준 계약 생성 및 모니터링을 위하여 WSLA[3]를 개발하였다. WSLA는 웹 서비스 사용에 있어서 웹 서비스 제공자와 요청자들의 의무사항들을 정의한 합의 문서이며, 서비스 수준 계약 환경의 특성을 고려하여 만들어졌다. WSLA는 <그림 1>과 같이 다음과 같은 요소로 구성되어 있다[3].



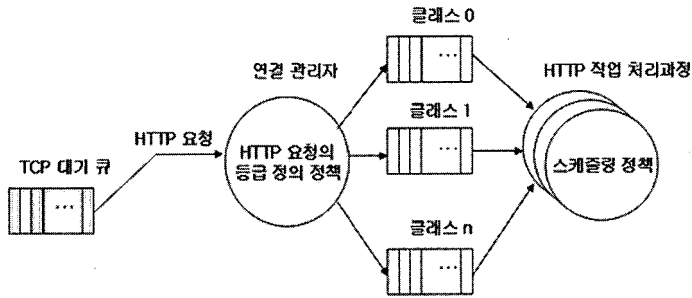
〈그림 1〉 WSLA 구조

2.3 웹 서버에서의 차별화 서비스

최근 인터넷의 발달과 대중화로 인하여 네트워크의 트래픽이 현저하게 증가하고 있다. 이로 인한 가장 큰 문제점은 서비스 종류에 따라 패킷 전달 기능을 차별화하지 않고 모든 패킷을 동등하게 최선을 다하여 처리한다는 것이다. 결과적으로 서비스의 품질을 보장하지 못하게 되었으며, 실시간 정보통신 서비스 등을 제공하는데 한계를 드러내고 있다. 이러한 문제점들을 해결하기 위하여 IETF에서 트래픽 제어 및 관리 기술인 차별화 서비스를 제안 하였다[10]. 차별화 서비스란 인터넷을 통하여 전달되어야 하는 패킷들을 사용자가 지정한 구분 방법에 따라 8~64가지의 서비스 유형으로 나누고, 각 서비스 유형에 대하여 각각의 패킷 교환기들이 수행해야 하는 처리 기능을 지정하여 인터넷 서비스 제공에 차별화가 이루어질 수 있도록 하는 것이다.

그러나, 이 차별화 서비스는 인터넷을 경유한 종단간 전송 품질의 보장이 되지 않으며, 통신망의 각 구간별 차별화 패킷 전송 처리 기능만이 구현되어 있다. 또한 최근에는 네트워크 지연시간보다 웹 서버에 부하가 집중되었을 경우 웹 서버에서의 지연시간이 더 커지기 때문에 차별화 서비스를 지원하기 위한 웹 서버에 대한 연구가 많이 진행되고 있다[8, 9, 11, 12]. 이러한 연구 중의 하나로 HTTP 요청에 대하여 분류 기준에 따라 클래스 별로 분류하고, 각 클래스에 따라 서비스 품질을 차별화하여 서비스할 수 있는 WebQoS의 구조를 제안하였다[9]. 〈그림 2〉는 이러한 방법의 구조를 나타내고 있다.

WebQoS는 〈그림 2〉와 같이 기존의 웹 서버가 FIFO 방식으로 HTTP 요청을 처리하는 대신 연결 관리자를 추가 하여 우선순위에 의하여 클래스를 구분한다. 이렇게 분류된 클래스에 저장된 HTTP 요청은 스케줄링 정책



〈그림 2〉 차별화 서비스를 지원하는 웹 서버의 구조[9]

에 실행되어지며, 이렇게 함으로서 차별화된 서비스 품질을 지원하는 방법이다.

본 논문에서는 WebQoS의 구조를 이용하여 개선된 스케줄링 정책을 제안한다.

3. 웹 서비스 품질 정보를 이용한 우선순위 할당 기법

3.1 사용되는 웹 서비스 품질 요소의 도출

웹 서비스 품질 요소에 대하여 [2]에서는 크게 성능 및 안정성 측면, 미들웨어 측면, 관리 가능성 측면, 상호운용성 측면 등으로 다양하게 분류하고 있다. 하지만, 우선순위를 결정하기 위하여 너무 많은 품질 요소를 반영하면, 이러한 품질 요소들을 실시간으로 측정함과 동시에 우선순위를 고려하기 위한 계산을 수행해야 하기 때문에 정확성 및 성능적인 측면에서 무리가 따른다. 이에 본 논문에서는 서비스 수준 계약에서 가장 많이 사용되며 모니터링 하기가 용이한 성능 및 안정성 측면의

품질 요소를 사용하기로 결정하였는데, 이 품질 요소는 사용자의 기대 요소를 잘 반영하고 있다. 다음은 고려되는 성능 및 안정성 측면의 품질 요소에 대하여 계산할 수 있는 식을 정의하였다.

• 성능 측면의 품질 요소

성능 측면의 품질 요소를 반영하여 우선순위를 부여함으로써 높은 성능을 요하는 서비스 사용자에게 차별화된 서비스를 제공할 수 있다.

- 응답 시간(Response Time) = 메시지 처리 시간의 총합 / 메시지 처리 수
- 처리량(Throughput) = 총 메시지 처리 수 / 총 처리 시간

• 안정성 측면의 품질 요소

안정성 측면의 품질 요소는 웹 서비스의 사용 가능성에 관한 품질 요소이며, 이 역시 차별화 서비스가 이루어져야 한다. 사용자 관점에서는 웹 서비스가 사용 가능한 상태에 있더라도 성능이 나쁘면 서비스가 가용하지 않다고 생각할 수 있다[1]. 또한, 그 측정에 있어

서도 타임아웃과 동시에 많은 서비스 요청이 있을 경우에 안정성 측면의 품질 요소는 떨어지게 된다. 그러므로 웹 서버에서 우선순위를 할당하는 요인으로 이러한 요인들을 감안함으로써 안정성 측면의 품질 요소에 대한 차별화 서비스에 영향을 미칠 수 있다.

- 접근성(Accessibility) = 특정 메시지에 대한 메시지 처리 비율(신뢰성)
- 신뢰성(Reliability) = 메시지 처리 수/요청된 메시지 수

위의 접근성에서 말하는 처리 비율은 신뢰성에서 계산하는 식(메시지 처리 수/요청된 메시지 수)과 같다. 하지만, 서버의 최대처리량을 넘어서게 될 경우에도 고급 사용자들의 처리 비율을 보장하기 위하여 접근성을 고려하여야 한다.

이와 같이 정의한 웹 서비스 품질 요소를 보장하기 위해서는 사용자와 제공자간의 서비스 수준 계약을 체결해야 한다. 일반적으로 서비스 수준 계약은 정량화할 수 있는 애플리케이션, 서비스 네트워크 등의 성능을 정의된 수준으로 보증하기 위한 서비스 제공자와 사용자간의 공식적인 계약이다. 하지만 아직까지 서비스 수준 계약에 대한 표준이 존재하고 있지 않기 때문에 본 논문에서는 IBM의 WSLA 1.0[3]을 사용하기로 하였다. 또한, 서비스 수준 계약은 아주 포괄적일 수도 있고 매우 상세화될 수도 있으며, 보통 계약 이행에 실패한 경우 서비스 제공자와 사용자가 취해야 하는 단계들을 포함하고 있다. 그러나 본 논문에서는 위에서 정의한 서비스 품질 요소에 대해서만 작성하는 것으로 한다. 즉, 본

논문에서는 서비스 품질 요소들에 대하여 최대한으로 만족하는 것을 주 목적으로 한다.

서비스 품질 요소들을 표현하기 위하여 WSLA 1.0의 <SLAParameter> 엘리먼트를 이용한다. 다음 <그림 3>은 위에서 정의한 서비스 품질 요소를 WSLA 1.0의 <SLAParameter>를 이용하여 표현한 것이다.

```

.....
<SLAParameter name = "AverageResponseTime"
                type = "float"
                unit = "seconds">
  <Metric>AverageResponseTime</Metric>
</SLAParameter>
<SLAParameter name = "Throughput"
                type = "integer"
                unit = "transactions/hours">
  <Metric>Throughput</Metric>
</SLAParameter>
.....
    
```

<그림 3> WSLA <SLAParameter>의 예

<그림 3>에서 정의된 서비스 품질 요소들에 근거하여 당사자들은 각자가 서로에게 제공해야하는 보증을 분명히 정의해야 한다. WSLA 1.0에서는 Service Level Objective와 Action Guarantees를 이용하여 두 가지 종류의 보증을 제공한다. Service Level Objective는 <그림 3>과 같이 SLA Parameter와 관련한 약속 즉, 당사자는 주어진 기간 동안 서비스의 특정 상태를 유지하겠다는 약속을 나타낸다. 어떤 당사자든 이 보증의 의무를 질 수 있는데, 주로 서비스 제공자 측이 전형적으로 이 역할을 수행해야 한다. Action Guarantees는 어떠한 동작을 수행하겠다는 약속이며

Service Level Objective의 위반 공지나 관리 작동 호출 등이 포함된다. 본 논문에서는 서비스 품질 요소들을 어느 정도까지 만족해야 하는지를 표기하기 위하여 Service Level Objective만을 사용하기로 하며, 이 값들을 이용하여 메시지를 분류하고 우선순위를 할당하는 데 사용한다. 다음 <그림 4>는 WSLA 1.0을 이용하여 Service Level Objective를 정의한 예제이다.

3.2 우선순위 할당 기법

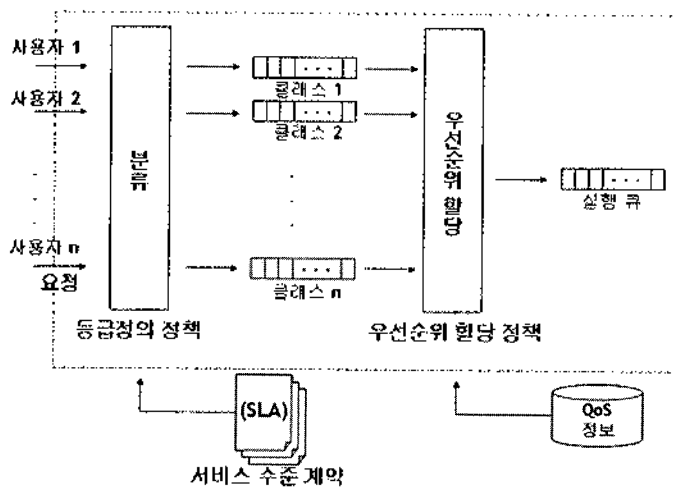
차별화된 웹 서비스를 지원하기 위한 웹 서버의 모델은 <그림 5>와 같다. 이 모델에서는 기본적으로 웹 서비스 사용자와 제공자간에 서비스 수준 계약이 체결되어 있어야 하며, 그 중에서 3.1절에서 언급한 웹 서비스 품질 정보에 대한 측정값이 있어야 한다. 웹 서비스 사용자들에 의하여 요청된 웹 서비스 메시

```

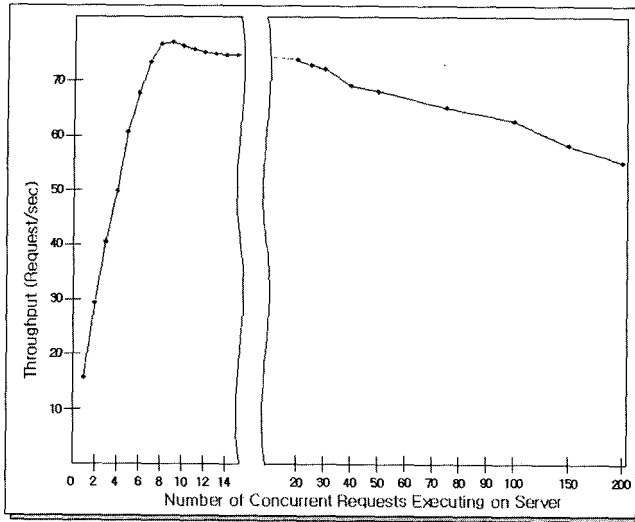
<Obligations>
  <ServiceLevelObjective name="Ex_SLO">
    <Obligated>ServiceProvider</Obligated>
    <Validity>
      <Start>2004-10-10T14:00:00.000-05:00</Start>
      <End>2004-12-31T14:00:00.000-05:00</End>
    </Validity>
    <Expression>
      <Implies>
        <Expression>
          <Predicate xsi:type="Less">
            <SLAParameter>AverageResponseTime</SLAParameter>
            <Value>1000.0</Value>
          </Predicate>
        </Expression>
        .....
      </Implies>
    </Expression>
  </ServiceLevelObjective>
</Obligations>
    
```

<그림 4> WSLA <ServiceLevelObjective>의 예

지는 분류 정책에 의하여 분류되며, 실행 큐에서는 최대 처리 성능을 보장할 수 있는 메시지 수에 한하여 처리할 수 있게 한다. 그 이



<그림 5> 차별화 웹 서버의 모델



〈그림 6〉 처리량 성능의 예

상의 요청이 발생하였을 경우에는 각각의 해당 버퍼 큐에서 대기하게 되며, 분류된 버퍼 큐에서 실행 큐로 보내어질 메시지는 우선순위 정책에 의하여 결정되어진다.

이 모델에서는 먼저 웹 서버가 처리할 수 있는 최적의 메시지 수를 제어할 수 있어야 한다. 웹 서버에 요청된 메시지가 증가하게 되면, 그 처리량은 포화 지점까지 다다르게 되고 응답시간 또한 현저하게 늘어나게 된다. 이는 포화 지점을 지나게 되면 성능은 리소스 충돌, 컨텍스트 변환 등으로 감소하게 되기 때문이다[5]. 그러므로 웹 서비스의 개시나 서비스 수준 계약 이전에 이러한 포화 지점을 알아내야 하며, 이를 이용하여 실행 큐에 적합한 메시지의 수를 정할 수 있어야 한다.

〈그림 6〉은 4장의 실험 환경에서 수행해야 할 서버의 처리량을 측정하는 것이다. 이 실험에서는 서버 상에서 동시에 실행되는 메시지

요청의 수 즉, 실행 큐의 길이가 9일 때 가장 좋은 성능을 보여주고 있다. 4장의 실험에서도 이 수치를 이용하여 실행 큐의 길이를 정하였다.

차별화된 웹 서비스를 지원하는 웹 서버를 구축하기 위하여 크게 각 메시지를 분류하는 모듈과 분류된 메시지들에 대하여 우선순위를 할당할 수 있는 모듈이 필요하다. 각 메시지를 분류하는 모듈은 서비스 수준 계약에 의하여 분류할 수 있으며, 이들의 구분은 사용자의 IP 주소나 URL 형태 등으로 구분할 수 있다. 우선순위를 할당하는 모듈은 축적된 웹 서비스 품질 정보를 이용하여 적용할 수 있다.

우선순위를 할당하는 모듈에서는 실행 큐가 포화상태가 되었을 때 각 분류된 큐에서 대기하고 있는 메시지들에 대하여 우선순위를 할당하여 우선순위가 가장 높은 메시지를 차례대로 실행 큐로 전달하여야 한다. 본 논

문에서는 이러한 우선순위를 책정하기 위하여 3.1절에서 도출한 웹 서비스 품질 요소를 모니터링한 값과 서비스 수준 계약에 의하여 체결된 품질 요소 값을 비교하여 전체적으로 서비스 수준 계약을 높은 수준에서 만족할 수 있도록 한다. 기본적으로 분류된 큐에서는 FIFO 방식이 적용되어 가장 먼저 들어온 메시지가 그 큐에서는 가장 먼저 서비스가 되어진다. 분류된 큐들에서 대기하고 있는 메시지들이 있을 때 각 큐에서 가장 먼저 들어온 메시지들에 대하여 비교 대상이 선정되며, 이 메시지들에 대하여 품질 정보 값을 산출하여 이 값이 가장 작은 메시자에게 가장 높은 우선순위가 할당된다.

응답 시간은 사용자가 품질 요소를 평가할 때 직접적으로 느낄 수 있는 요소이며, 서비스 수준 계약에서도 높은 비중을 두고 계약을 체결하는 경우가 많다. 이 응답 시간을 적용하기 위하여 다음과 같은 (1)의 식을 적용하여 값을 산출할 수 있다.

$$V_{RT} = RT_{SLA} - QT - RT_{mean} \quad \text{--- (1)}$$

(1)의 식에서 RT_{SLA} 는 서비스 수준 계약에 서로 상호간에 계약된 평균 응답 시간의 값이며, QT 는 큐에서 그 메시지가 대기한 시간, RT_{mean} 은 그동안 측정된 평균 응답 시간을 나타낸다. 기본적으로 (1)의 식을 이용하여 V_{RT} 값이 가장 적은 즉, 만족해야 할 응답시간의 여유가 가장 조금 남아 있는(V_{RT} 가 가장 작은) 메시자에게 우선순위를 결정할 수 있으며, 이는 EDF의 방식과 유사한 기능을 한다.

EDF(Earliest Deadline First) 방식은 마감

시간이 가까운 메시자에게 우선적으로 우선순위를 부여하는 방식이다. 대기하고 있는 메시지들 중에서 마감시간이 가장 짧은 메시지를 선택하여 실행 큐로 보내어, 각 클래스의 우선순위가 시간의 흐름에 따라 동적으로 변화한다. 이 방식은 마감시간을 예측할 수 있고 응답시간을 다른 방식에 비하여 잘 만족할 수 있다는 장점이 있다.

하지만, 본 논문에서는 응답 시간 이외에도 다른 품질 요소를 고려하여 응답 시간의 만족률과 함께 다른 품질 요소에 대한 전체적인 만족률을 향상시키기 위하여 해당 품질 요소의 만족률을 계산한다. 다음 (2)의 식은 해당 메시지의 처리량에 대한 만족률(V_T)이다.

$$V_T = T / T_{SLA} \quad \text{--- (2)}$$

(2)의 식에서 T_{SLA} 는 서비스 수준 계약에 서로 상호간에 계약된 처리량의 값이며, T 는 그동안 측정된 처리량을 나타낸다. 또한, 접근성과 신뢰성에 대한 만족률(각각 V_{Ac} , V_R)을 다음과 같이 구할 수 있다.

$$V_{Ac} = Ac / Ac_{SLA} \quad \text{--- (3)}$$

$$V_R = R / R_{SLA} \quad \text{--- (4)}$$

위의 식에서 Ac_{SLA} 와 R_{SLA} 는 서비스 수준 계약에 서로 상호간에 계약된 각각 접근성, 신뢰성의 값이며, Ac 와 R 은 그동안 측정된 접근성과 신뢰성을 나타낸다.

(1) ~ (4)의 식에서 구해진 값들을 이용하여 우선순위에 할당하는데 사용하는 값을 다음과 같이 계산할 수 있다.

$$V_P = V_{RT} * [(V_T + V_R) / 2] \quad \text{--- (5)}$$

각 대기 큐에서 대기하고 있는 가장 먼저 들어온 메시지들에 대하여 (5)의 식을 이용하여 V_P 의 값을 계산 및 비교하여 가장 값이 작은 메시지에 높은 우선순위를 부여할 수 있다. 이는 응답시간을 기반으로 하며, 그동안 측정된 처리량과 신뢰성의 값과 서비스 수준 계약의 값을 비교하여 시간이 경과함에 따라 서비스 수준 계약의 만족도가 높아지게 된다.

하지만, (5)의 식을 그대로 사용하기에는 무리가 따른다. V_P 의 값이 음수일 경우에 V_P 의 값 중 최소값으로 우선순위를 정하면 우선 순위가 뒤바뀔 수 있다. 그렇기 때문에 V_P 의 값은 다음과 같이 수정되어야 한다. 여기서 TimeOut의 값은 V_{RT} 의 값을 양수로 만들기 위하여 사용하며, V_{RT} 에 TimeOut을 더한 값은 다른 V_T, V_R 의 값과 비교하기 위하여 ln의 값을 취한다.

$$V_P = \ln(V_{RT} + \text{TimeOut}) * [(V_T + V_R) / 2] \quad \text{--- (6)}$$

(6)의 식을 이용하면, 서버가 처리할 수 있는 범위 안에서는 각 요청자들의 서비스 수준 계약을 모두 만족할 수 있다. 하지만, 정해진 용량을 벗어나게 되면 즉, 요청수가 처리할 수 있는 수보다 더 많이 들어오게 되면 어느 쪽의 서비스를 포기해야 한다. 이 때, 서비스 수준 계약에 접근성의 값을 계약한 서비스 요청자를 고급사용자라 간주하며, 서버가 최대 성능을 보일 수 있는 초당 요청수를

threshold값으로 설정한다. 그리고 접근성의 값을 계약한 서비스 요청자의 메시지에는 다음 (7)의 식을 적용하며, 일반 요청자의 메시지는 (8)의 식을 사용한다. 이는 threshold값이 초과되었을 때에 고급사용자에게는 최대한의 성능을 보장하기 위하여 신뢰성의 값을 사용하며, 일반사용자에게는 최소한의 성능을 보장하기 위하여 처리량의 값을 사용한다.

$$V_P = \ln(V_{RT} + \text{TimeOut}) * V_{Ac} \quad \text{-- (7)}$$

$$V_P = \ln(V_{RT} + \text{TimeOut}) * V_T \quad \text{-- (8)}$$

본 논문에서 (6), (7), (8)의 식을 우선순위를 할당하는 식으로 제안하고 있지만, 응답시간을 제외한 다른 요소가 빠져있다면 그 요소를 배제한 식을 사용할 수 있다.

4. 실험 및 평가

4장에서는 실험을 통하여 본 논문에서 제안하고 있는 웹 서비스 차별화를 위한 우선순위 할당 기법과 EDF 방식과 비교하여 그 유용성을 검증해 보고자 한다.

4.1 실험 시나리오

실험은 네트워크 상황을 배제하고 LAN 환경에서 애플리케이션 수준의 시뮬레이션으로 이루어진다. 각각 하나의 웹 서비스 처리 시간은 약 0.06초 정도가 소요되게 하였으며, 신뢰성, 응답시간, 처리량 등을 측정하였다.

이 실험에서는 웹 서비스 메시지를 세 가지

```

.....
<Obligations>
.....
  <Expression>
    <Predicate xsi:type="Less">
      <SLAParameter>
        AverageResponseTime</SLAParameter>
        <Value>200</Value> <! -- 2sec -->
      </Predicate>
    </Expression>
  <Expression>
    <Predicate xsi:type="Greater">
      <SLAParameter>
        Throughput</SLAParameter>
        <Value>30</Value>
      </Predicate>
    </Expression>
  <Expression>
    <Predicate xsi:type="Less">
      <SLAParameter>Availability_CurrentDown
        Time</SLAParameter>
        <Value>0.01</Value>
      </Predicate>
    </Expression>
  <Expression>
    <Predicate xsi:type="Greater">
      <SLAParameter>
        Accessibility_Transaction</SLAParameter>
        <Value>0.95</Value> <! -- 95% -->
      </Predicate>
    </Expression>
  <Expression>
    <Predicate xsi:type="Greater">
      <SLAParameter>Reliability_Transaction
        Rate</SLAParameter>
        <Value>0.95</Value> <! -- 95% -->
      </Predicate>
    </Expression>
  <EvaluationEvent>NewValue</EvaluationEvent>
</ServiceLevelObjective>
</Obligations>
.....

```

〈그림 7〉 클래스1의 WSLA

로 구분하였으며, 다음 〈그림 7〉은 이 실험에서 사용하는 WSLA의 일부분을 나타내고 있다.

〈그림 7〉에서 나타내듯이, 클래스1로 구분되는 메시지에 대하여 2초 이내의 응답시간, 30개 이상의 처리량, 95% 이상의 신뢰성을 계약하고 있다. 이 실험에서는 클래스1 > 클래스2 > 클래스3의 순으로 품질을 보장하는 계약을 가정한다. 클래스2로 구분되는 메시지에 대해서는 각각 2.5초, 25개, 90%, 그리고 클래스3으로 구분되는 메시지에 대해서는 각각 3.5초, 20개, 80%의 값을 계약하는 것으로 가정하며, 〈그림 7〉과 같은 종류의 WSLA를 작성할 수 있다.

4.2 결과 및 분석

〈그림 8, 9, 10〉에서 나타내듯이 이 실험환경에서 서비스 수준 계약을 모두 만족할 수 있는 최대성능치는 초당 요청수가 120일 때라고 할 수 있다. 〈그림 8〉에서 EDF 방식과 제안 방식의 응답시간이 편차를 보이고 있지만, 두 방식 모두 초당 요청수가 120일 때에 클래스1, 2, 3의 응답시간을 잘 만족하고 있다. 하지만, 신뢰성과 처리량의 경우에는 EDF 방식이 초당 요청수가 120일 때, 서비스 수준 계약을 잘 만족시키지 못하고 있다. 이는 신뢰성과 처리량에 관하여 EDF 방식의 경우 서비스 수준 계약을 모두 만족하기 위해서는 초당 요청수가 120을 넘지 않아야 한다는 뜻이며, 반면에 제안 방식의 경우 초당 요청수가 120일 때에도 서비스 수준 계약을 모두 만족할 수 있다는 것을 나타낸다.

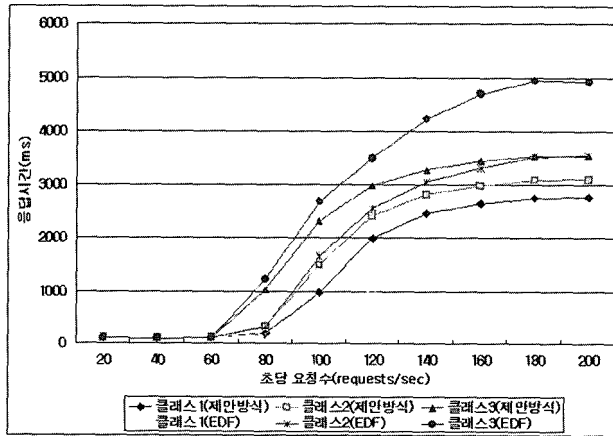
초당 요청수가 120이 초과되었다는 것은 이

서버의 최대 처리량을 넘어선 것으로 볼 수 있다. 그렇기 때문에 모든 서비스 수준 계약을 만족 시킬 수 없으며, 고급사용자를 위하여 일반사용자의 요청을 어느 정도 무시할 수 밖에 없다. 다음 <그림 11>은 본 논문에서 제안하는 기법에서 접근성을 고려하지 않은 실험과 고려한 실험의 신뢰성에 관한 결과이다. 이렇게 접근성을 이용하여 고급사용자의 성

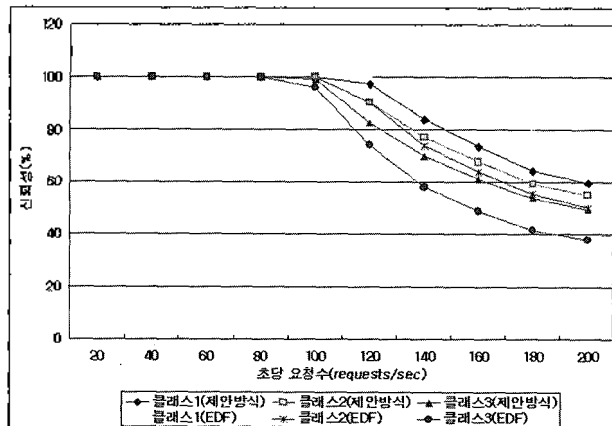
능을 최대한으로 보장할 수 있다.

5. 결론 및 향후 연구

최근에 많은 기업들은 XML 메시지 기반의 분산 환경에서 이기종 간의 표준으로 자리 잡고 있는 웹 서비스를 도입하여 B2C 뿐만 아니



<그림 8> EDF 방식과 제안 방식의 응답시간



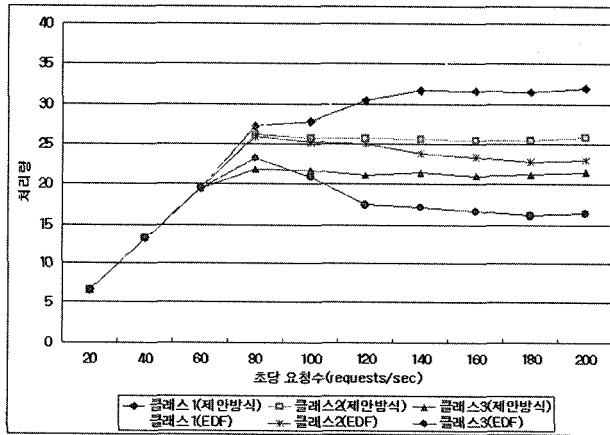
<그림 9> EDF 방식과 제안 방식의 신뢰성

라 B2B에 거쳐서 기업과 기업간 동적인 비즈니스를 수행하고 있다. 이러한 웹 서비스가 성공적으로 비즈니스를 수행하기 위해서는 차별화된 서비스를 제공할 수 있어야 한다.

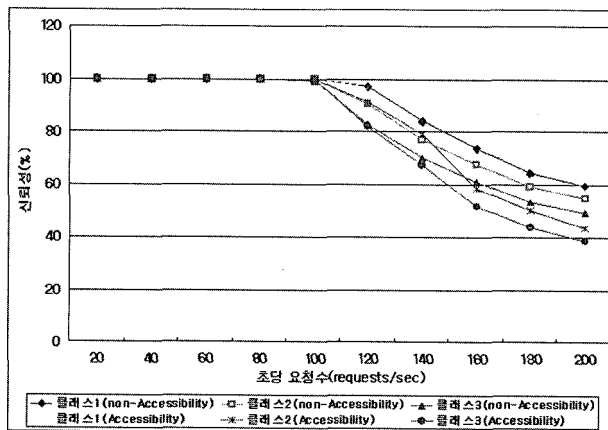
그러나 네트워크 수준에서의 차별화 서비스는 종단간 전송 품질의 보장이 되지 않지만 다행히도 네트워크 환경이 점점 더 좋아지고 있는 추세이다. 또한, 웹 서비스에서는 XML

을 이용하여 메시지를 전달하기 때문에 이 XML을 파싱해야 하는 부담이 있고, 현재 대부분의 웹 서비스는 웹 서버를 이용하여 통신을 하고 있다.

이에 본 논문에서는 네트워크 수준이 아닌 애플리케이션 수준의 웹 서버 상에서 차별화된 웹 서비스를 제공하는 데 필요한 우선순위 할당 기법을 제안하였다. 이렇게 하기 위하여



〈그림 10〉 EDF 방식과 제안 방식의 처리량



〈그림 11〉 접근성을 고려한 신뢰성

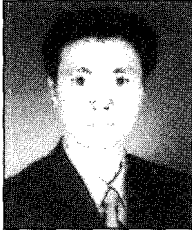
먼저 웹 서비스에서 주로 사용하는 웹 서비스 품질 요소들을 분석하였고, 우선순위를 할당 하는데 필요한 웹 서비스 품질 요소를 도출하였다. 이러한 품질 요소들과 IBM의 WSLA를 이용하여 웹 서버에서 메시지들의 우선순위를 할당하는 기법을 제안하였다.

향후 연구로는 본 논문에서는 네트워크 속도를 고려하지 않고 있기 때문에 DiffServ와 같은 네트워크 수준에서의 차별화 서비스와 연동할 수 있는 방안이 필요하다.

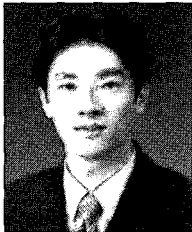
참 고 문 헌

- [1] 박상근. "서비스 저하요소를 고려한 웹 서비스 가용성 측정모델", 전남대학교 대학원 전산학과 박사학위논문, 2003.2
- [2] 한국전산원. "웹 서비스 품질관리 동향 및 도입전략 연구", 2003.12
- [3] A. Keller and H. Ludwig, "The WSLA Framework: Specifying and Monitoring of Service Level Agreements for Web Services", IBM, 2003.6
- [4] Anbazhagan Mani and Arun Naqarajan, "Understanding quality of service for Web services", IBM, 2002.1
- [5] Asit Dan, Heiko Ludwig, and Giovanni Pacifici, "Web services differentiation with service level agreements", White Paper, IBM, 2003.5
- [6] Daniel A. Menasce, "QoS Issues in Web Services", IEEE Internet Computing, 2002.12
- [7] Liangahao Zeng, Boualem Benatallah, Malon Dumas, Jayant Kalagnanam, and Quan Z. Sheng, "Quality Driven Web Services Composition", www2003, 2003.5
- [8] N. Vasiliou and H. Lutfiyya, "Providing a Differentiated Quality of Service in a World Wide Web Server", Proc. Of the Performance and Architecture of Web Servers Workshop, Santa Clara, California USA, pp. 14-20, 2000.6
- [9] R. Bhatti and R. Friedrich., "Web Server Support for Tiered Services", IEEE Network, 13(5):64-71, 1999.9
- [10] S. Blake, D. Black, M. Carlson, E. Davies, Wang Z. and W. Weiss.m "An architecture for differentiated services", IETF RFC 2475, 1998.12
- [11] Xiangping Chen and Prasant Mohapatra, "Performance Evaluation of Service Differentiating Internet Servers", IEEE Trans. on Computers, 51(11):1,368-1,375, 2002.11
- [12] Xiaobo Zhou, Yu Cai, Ganesh K. Godavari, and C. Edward Chow, "An Adaptive Process Allocation Strategy for Proportional Responsiveness Differentiation on Web Servers", IEEE International Conference on Web Services(ICWS 2004), pp. 142-149, 2004.7

저 자 소 개



김동준 (E-mail : djkim@tysystems.com)
2002. 강원대학교 통계학과(이학사)
2005. 중앙대학교 대학원 컴퓨터공학과(공학석사)
2005. ~ 현재 동양시스템즈 기술연구소 연구원
관심 분야 웹 서비스, 시맨틱 웹, ebXML



이상규 (E-mail: sklee@archi.cse.cau.ac.kr)
2005. 중앙대학교 컴퓨터공학과(공학사)
2005. ~ 현재 중앙대학교 컴퓨터공학과 석사과정
관심 분야 웹 서비스, 그리드 컴퓨팅



한상용 (E-mail: hansy@cau.ac.kr)
1975. 서울대학교 공과대학(공학사)
1984. Minnesota 공과대학(공학박사)
1984 ~ 1995. IBM 책임연구원
1995 ~ 현재 중앙대학교 컴퓨터공학과 교수
관심 분야 웹 서비스, 시맨틱웹, 정보 검색