QoS 를 보장하는 분산 네트워크 스토리지의 구조 연구

홍승욱*, 안종석**, 박찬익†
*,**동국대학교 컴퓨터공학과
†포항공과대학교 컴퓨터공학과
e-mail : {swhong, jahn}@dgu.edu, cipark@postech.ac.kr

Surveying Distributed Network Storage Architectures Providing Quality of Service

Seung-Wook Hong*, Jong-Suk Ahn**, Chan-Ik Park †
*,** Dept. of Computer Engineering, Dong-Guk University
† Dept. of Computer Science and Engineering, POSTECH

요 약

네트웍이 고속화 됨에 따라 대용량 데이터를 실시간으로 전송/처리/저장하는 응용 분야가 등장하고 있다. 또한 전송 지연을 감소하고 신뢰도를 향상하기 위해서 많은 스토리지(storage) replica 들이 네트웍에 설치되었다. 이러한 환경에서 응용 프로그램이 요구하는 성능을 만족하기 위해서는 네트웍 자원 뿐만 아니라 종단간 자원인 저장 매체의 선택/예약/관리를 함께 고려하는 통합적 분산 네트웍 스토리지(DNS: Distributed Network Storage) 시스템이 필요하다. 이러한 DNS 에서 QoS 를 제공하는 기존의 방식들은 크게 데이터그램(datagram) QoS, 클래스(class)별 차등적인 서비스를 제공하는 DiffServ QoS, 그리고 각 플로우(flow) 별 개별적인 서비스를 제공하는 IntServ QoS 로 세분된다. 본 논문에서는 기존에 제안된 방식들을 비교 분석하여 각각의 장단점을 기술하며 향후 연구 문제들을 살펴본다.

1. 서론

인터넷이 고속화 됨에 따라 대용량 데이터를 실시간으로 저장하거나 읽어오는 응용 프로그램이 등장하게 되었다. 즉, 대용량의 데이터를 실시간으로 네트웍을 통해 전송/처리/저장을 하는, 다중 데이터베이스 시각화 프로그램, 원격 진료 시스템, GRID 등의 응용들이 인터넷상에서 구현되고 있다.

이러한 응용 프로그램들이 요구하는 성능을 제공하기 위해서는 자원들, 즉 전송 대역폭, CPU, 그리고 저장 매체들을 알맞게 배분하고 관리하는 것이 필요하다. 더욱이 전송 지연을 감소하고 신뢰도를 향상하기 위해 설치된 replica 들의 위치와 그들의 이용 가능성을 고려하여 저장 매체를 선택해야 할 필요가 생겼다.

종래에는 이러한 자원 배분문제는 서로 관련 없이 개별적으로 연구/구현되어 왔다. 즉 네트웍 대역폭은 네트웍 QoS 분야에서, CPU 와 저장 매체의 배분 문제는 스케쥴링(scheduling) 연구 분야에서 각각 활발히 연구되었다. 그러나, 개별적인 자원의 배분과 관리 방식에 의해서는 대용량 데이터를 실시간으로 전송/처리/저장해야 하는 응용 프로그램들의 요구를 충족시키기 어렵고, 또한 replica 들을 효율적으로 이용할 수 없는 문제점을 가지고 있다. 일례로 주어진 종단간 지연을 만족하기 위해서는 이 종단간 지연을 네트웍 전송 지연, 처리 지연, 그리고 저장 지연으로 알맞게 나누어야 하는 데, 이를 위해서는 알맞은 replica 와 이 replica 로의 네트웍 통로를 결정하는 것이 필요하다.

WAN 상의 분산 스토리지 replica 들을 이용하여 통합적인 DNS QoS 를 제공하기 위한 연구들은 지원하는 응용 프로그램의 특성에 따라 크게 세가지로 분류된다. 첫째는 차별적인 서비스를 제공하지 않고 모든 자원을 공유하는 데이터 그램 QoS 을 제공하는 것이다. 이러한 방식은 다시 파일을 공유하는 분산 파일 시스템과 디스크를 공유하는 분산 스토리 시스템으로 분류된다. FTP mirroring, NAS(Network Attached Storage), Peer-to-peer 스토리지, CDN(Content Distribution Network) 등이 전자에 속하며, SAN(Storage Area Network)과 IPS(IP Storage)등이 후자에 속한다.

둘째는, 기존의 DiffServ 방식을 이용한 계층별 서비스를 제공하는 방식으로 응용 프로그램의 특성에 따라 미리 지정된 서비스 클래스 중에서 알맞은 클래스를 선택하는 방식이다. 이러한 방법 또한 데이터그램 QoS 와 마찬가지로 완전한 QoS 보장은 불가능하며 통계적인 보장만을 할 수 있다. 마지막은 InitServ 에 근

간하여 응용 프로그램이 요구하는 QoS 를 개별적으로 보장하는 InitServ QoS 방식이다.

이러한 세가지 방식은 각각 문제점을 가지고 있는데, 데이터그램 방식은 어떠한 QoS 도 보장할 수 없다는 문제를 가지고 있으며, DiffServ 방식은 정해진 서비스 클래스중에서 주어진 응용 프로그램에 알맞은 클래스를 선택하는 문제가 있고, InitServ 방식은 복잡하기 때문에 확장성에 문제가 있다. 따라서 이 세가지 방식은 각 응용 프로그램의 특성에 따라 선택적으로 사용될 것으로 예상된다. 본 논문에서는 이들 세가지 방식의 구현 방법, 장단점 그리고 향후 연구 과제들을 알아본다.

본 논문의 구성은 다음과 같다. 2 장은 기존의 단일(best-effort) 서비스만을 제공하는 DNS 시스템들에 대해서 3 장은 QoS 를 제공하는 DNS 를 DiffServ 와 IntServ 구현 방식을 요약하고, 4 장은 이들 세가지 방식의 장단점을 바탕으로 DNS QoS 의 향후 연구 과제를 논의하며, 마지막으로 5 장에서는 결론을 기술한다.

2. QoS 를 제공하지 않는 DNS

단일(best-effort) 서비스만을 제공하는 DNS 시스템은 입출력 단위를 기준으로 파일 단위의 공유 스토리지 시스템과 디스크 블록단위 공유 스토리지 시스템으로 구분할 수 있다.

2.1 블록 단위의 DNS

블록 단위의 DNS 로는 SAN 과 IPS 가 대표적인데, <그림 1>에서는 이들 블록 단위의 DNS 시스템의 프로토콜 스택을 보여주고 있다. DAS(Direct Attached Storage), SAN, iSCSI 은 로컬(local) 파일시스템을 거쳐 원격의 스토리지와 직접 혹은 IPS 를 통해 블록 단위의 입출력을 한다.

<그림 1>에서 ①DAS 는 로컬의 디스크를 접근하는 일반적인 구조를 나타내며, ②SAN 은 다양한 스토리지들을 기존의 시스템 버스가 아닌 광(fiber) 채널 버스로 연결하는 스토리지 네트웍이다. 광채널 스위치를 사용하면, 여러 서버들이 같은 SAN 에서 스토리지를 공유할 수 있어 확장과 관리가 용이하다.

로컬 SAN 을 확장하여 IP 네트웍를 연결 버스(bus)로 사용하는 다양한 IPS 프로토콜들이 등장하였다. 대표적으로 iSCSI[1], FCIP(Fiber Channel IP)[2], HyperSCSI[3] 등이 있는데, FCIP 는 SAN 영역(island)간의 게이트웨이 프로토콜로서 분산된 SAN 영역을 IP 네트워크를 통해 연결할 수 있도록 하며, iSCSI 프로토콜은 <그림 1>의 ③에서와 같이 기존의 TCP/IP 계층 위에 입출력을 위한 추상 프로토콜인 SCSI (Small Command System Interface) 계층을 추가하여 IP 네트워크를 통해 블록단위 입출력을 한다. 또한 HyperSCSI 는 IP 계층위에 TCP 대신 자체 전송계층과 SCSI 계층을 설계하여 기존의 연결지향적 전송계층구조를 입출력 특성에 맞게 설계한 프로토콜이다[4].

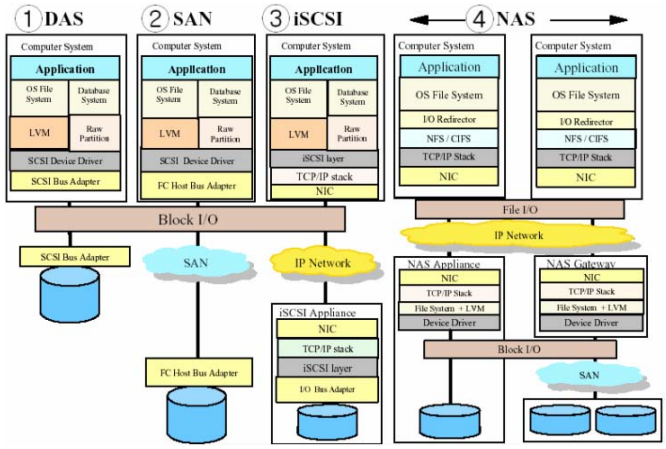

그림 1 SAN, iSCSI, NAS 구조

SAN 과 SAN 을 지원하는 IPS 은 향후 대용량 스토리지를 구현하기 위한 대표적인 프로토콜이나, 입출력을 위해 인터넷을 데이터 버스로 사용하기에는 지연이 크고 신뢰성이 떨어지는 문제가 있다. 네트워크 지연과 스토리지 자원을 동시에 고려한 QoS 메커니즘의 뒷받침 없이는 iSCSI 과 같이 인터넷을 경유한 블록 단위 입출력은 실현 가능성이 없는 것으로 예상된다.

2.2 파일 단위의 DNS

파일 단위의 DNS 로는 FTP mirroring, NAS, CDN, Peer-to-Peer 스토리지, 그리고 IBP (Internet Backplane Protocol)[5] 등이 있다. FTP mirroring 과 NAS 는 사용자가 미리 지정된 원격에 replica 를 설치하는 방식이다. NAS 방식의 네트워크 스토리지는 <그림 1>의 ④과 같이 원격 스토리지 서버 혹은 NAS 게이트웨이의 파일 시스템을 이용하여 파일 단위로 입출력을 수행하는 방식이며, 대표적인 프로토콜로는 NFS(Network File System)와 CIFS(Common Internet File System)가 있다.

NAS 는 하나의 파일 시스템 서버를 이용해 스토리지에 접근하는 중앙 집중적인 구조를 가지므로 관리 비용이 낮고 다양한 시스템에 적응적이지만 스토리지 서버에 입출력 부하가 집중되고 확장성이 떨어지는 단점이 있다. FTP mirroring 나 NAS 방식에서는 자원의 위치가 미리 결정되어 있지만, CDN 은 여러 곳에 replica 를 설치하고 DNS 서비스를 이용하여 지정된 URL 을 수신자와 가장 근접한 replica 로 동적으로 매핑(mapping)하여 스토리지 접근 시간을 줄인다.

P2P 스토리지는 동적으로 생성되는 각 호스트의 스토리지를 공유하는 방식으로 필요한 자원을 찾기 위해 세가지 방식을 사용한다. 첫째는 하나의 중앙 서버를 사용하는 방법이며, 둘째는 여러 개의 분산 서버를 이용하는 방식이며, 마지막은 일종의 가상 네트웍을 구성하고 이 가상 네트웍에 검색 패킷을 플러딩(flooding)하여 필요한 자원을 찾는 방식이다.

P2P 스토리지가 이러한 세가지 방식으로 파일을 공유하는 것을 목적으로 하는 반면 IBP 는 실제 디스크를 공유하는 방식이다. 즉 하나의 파일이 IBP 에서는 네트워크에 분산된 여러 개의 디스크에 존재할 수 있다. 이러한 기능을 수행하기 위해 IBP 는 <그림 2>에

서 어두운 색으로 표시한 프로토콜 스택을 구현하고 있다. ③IBP 층은 분산된 디스크들의 위치와 디스크의 상세 정보를 관리하여 분산 디스크들을 하나의 바이트 배열로 상위 계층에 제공하는 일종의 IP 와 같은 계층이다. 이러한 정보를 바탕으로 상위의 ①,② exNode 층은 Linux 에서 파일을 대표하는 inode 를 구성하는 것과 같은 기능을 수행하는 것이다. 즉, 하위 IBP 계층이 바이트 배열을 제공하고, exNode 는 이를 상위 응용(Application)계층에게 파일단위로 제공한다.

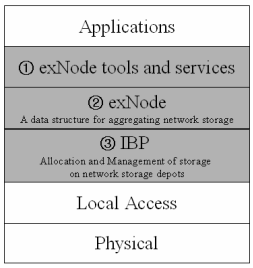

그림 2 IBP 프로토콜 스택

3. QoS 를 제공하는 DNS

WAN 상의 분산 스토리지 QoS 를 구현하기 위해서는 <그림 3>에서 표현한 것 과 같은 네 가지의 기본적인 기능들이 구현되어야 한다[6]. 첫째는 주어진 전송/처리 데이터의 특성과 요구하는 QoS 를 알맞게 표현하는 계수들(service specification)이며, 둘째는 요구하는 성능을 제공하는 네트웍 통로와 저장 매체를 발견(resource mapping)해야 하며, 셋째는 발견된 자원들을 예약/수락(admission control)해야 하고, 넷째는 약속된 QoS 를 제공(resource management)하기 위해 자원을 관리하는 것이다. 이러한 네 가지 관점에서 다음 두 방식, 차등서비스 와 통합서비스를 분석한다.

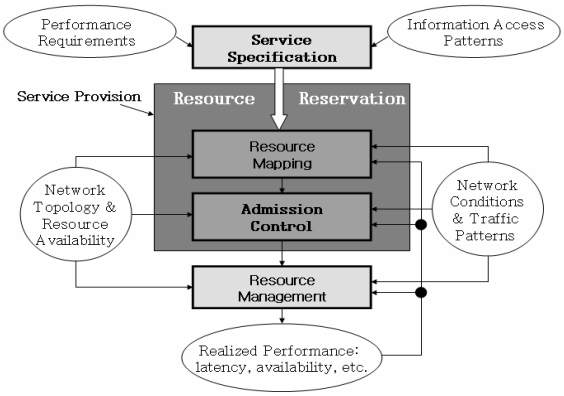

그림 3 QoS 를 보장하는 DNS 를 위한 기능의 흐름

3.1 차등(differentiated) 서비스

이 서비스는 응용 프로그램에게 미리 정해진 서비스 클래스에 따라 차등적 서비스를 보장하는 방식이다. GARA (Globus Architecture Resource Allocation)[7] 는 이 범주에 속하는 대표적인 방식으로, 데이터 그리드(Data Grid)의 QoS 를 위해 제안된 모델이다. 데이터 그리드는 미디어 (media) 데이터와 같이 지터(jitter)에 대한 영향은 적지만 대용량의 데이터 전송을 위한 대역폭 보장을 요구하는 응용 프로그램을 지원하기 위한 구조를 제공한다[8].

GARA 에서는 각 도메인마다 제공하는 서비스 클래스를 SLA (Service Level Agreement)로 제공하며 각 응용 프로그램은 알맞은 서비스 클래스를 선택한다. <그림 4>는 이러한 SLA 의 하나의 예를 보여주고 있는데, CPU 속도와 개수, 메모리 크기, 네트웍 대역폭과 손실율을 명시하고 있다[9].

<그림 5>는 GARA 에서 응용 프로그램이 ④원격 API(Application Program Interface)를 이용하여 자원을 예약하는 구조를 보여주고 있다. 사용자는 적당한 SLA 를 선택한 후에 각 도메인에 위치하는 ①차등 네트워크 자원관리자(DiffServ resource manager)와 ②분산 실시간 CPU 스케쥴러 (DSRT: Distributed Soft Real-Time) ③분산 병렬 스토리지 시스템 (DPSS: Distributed Parallel Storage System)에게 네트웍, CPU, 스토리지 자원의 예약을 요청한다. 각 자원관리자는 요청한 예약을 수행할 수 있는 지를 계산하고 가능하다면 요청자에게 수락 메시지를 전달하고, 라우터(router)와 해당 종단 호스트, 그리고 스토리지 서버에서 해당 자원을 예약하게 된다. 송신자는 패킷에 할당된 클래스를 마킹(marking)하여 데이터를 전송하게 되며, 라우터와 저장 매체는 WFQ (Weighted Fair Queuing)과 같은 알맞은 스케쥴링 기법을 이용하여 DiffServ 서비스를 제공한다.

```
<serviceSpecific>
        ...
   <cpuQoS unit="ghz" value="1">4</cpuQoS>
   <memoryQoS unit="mb">64</memoryQoS>
   <networkQoS>
      <sourceIP unit="raw">192.200.168.33</sourceIP>
      <destIP>135.200.50.101</destIP>
      <bandwidth unit="mbps">10</bandwidth>
      <packetLoss type="lessthan">10</packetLoss>
   </networkQoS>
</serviceSpecific>
```

그림 4 SLA 예제

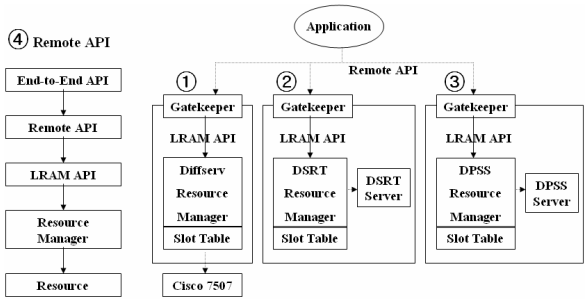

그림 5 GARA QoS 구조

3.2 통합(integrated) 서비스

통합 서비스는 특정 응용 프로그램이 요구하는 자원을 개별적으로 보장해 주는 방식으로, <그림 6>은 가능한 몇 개의 서비스 예를 보여준다[6]. 이 예에서 알 수 있듯이 요구하는 QoS 는 전송 지연, 저장 매체의 양, 사용 시간, 필요 replica 의 개수, 신뢰성 등의 수치로 표현되며 트래픽 프로파일은 토큰 버킷 깊이와 평균 전송 속도로 표현된다. 마지막으로 보장 방식에 따라 절대적 보장과 통계적 보장으로 구분된다.

이 방식에서 자원 검색과 예약은 RSVP[10]방식을 채택할 수 있다. 즉 저장 매체를 가지고 있는 호스트

는 RSVP 로 자신의 존재를 임의의 멀티캐스트 주소를 이용해서 전 라우터에게 알려 주며, 이때 사용자는 임의의 중앙 브로커(broker)를 통해 알맞은 저장 매체와 연관된 멀티캐스트 주소를 알아내어 이 주소에 예약 메시지를 보내 가입하게 된다. 가입한 후에는 사용자는 저장매체로부터 데이터를 가져오거나 저장을 할 수 있게 되며, 이때 라우터는 약속한 QoS 를 제공하기 위해 다양한 스케쥴링 방식을 채택할 수 있다. 그러나, RSVP 방식은 입출력의 부하가 높고, 양방향에 대한 예약을 독립적으로 수행하는 단점이 있다[11].

	Service	Description (traffic profile, performance requirements)
#1	Deterministic	1GB storage capacity for 1 hour, 100ms maximum latency
#2	Average	1GB storage capacity for 1 hour, 50ms average latency
#3	Combination	1GB storage capacity for 1 hour, (50ms average latency, 100ms worst case latency)
#4	Stochastic	1GB storage capacity for 1 hour, Probability[latency > 100ms] ≤ ε
#5	Geographic	1GB storage capacity for 1 hour, 100ms latency bound for all receivers in specific domain or region, or to specific set of receivers
#6	Budget-constrained	1GB storage capacity for 1 hour, minimizing worst-case latency, subject to budget constraint of no more than K replicas
#7	Placement-oriented	1GB storage capacity for 1 hour, at N specific nodes
#8	Advance reservation	1GB storage capacity from 2330hr, December 31 1999 to 0029hr, January 1 2000, 100ms latency bound

그림 6 네트워크 스토리지 QoS 의 서비스 계수 예제

4. 향후 DNS QoS 연구 방향

DNS 에서의 지금까지 파악한 향후 연구 문제들은 다음과 같이 요약할 수 있다. 첫째, QoS 요구와 전송 트래픽 표현 방식에 대한 연구가 계속 진행될 것이다. <그림 6>은 하나의 예를 보여주고 있는 데, 아직은 응용 프로그램들의 특성이 파악되지 않아, 스토리지 QoS 를 표현하는 계수들이 결정되지 않았다. 네트웍 QoS 는 전송 지연, 지터, 그리고 패킷 손실률로 정의되는 데, 스토리지 QoS 는 이들 계수이외에도 저장 매체 용량, 신뢰도 등이 첨부되어야 할 것이다. 또한 스토리지 응용 프로그램의 저장 매체 접근 방식을 전송 트래픽 량으로 변환하는 문제도 연구되고 있다 [12]. 마지막으로 DiffServ 의 경우에는 응용 프로그램의 QoS 요구와 클래스를 매핑하는 문제가 있다.

둘째, 수많은 replica 와 네트웍 통로 중에서 응용 프로그램이 요구하는 QoS 를 만족하는 자원들을 선택하는 문제가 연구되어야 할 것이다. 기존의 네트웍 QoS 에서는 알맞은 통로를 발견하기 위해, 목적지가 정해진 상태에서 모든 통로를 시도하는 방법을 사용하고 있으나, 목적지 즉 저장 매체가 많은 경우에 이러한 방법의 복잡도는 매우 높아진다. 또한 QoS 가 전송 지연으로 표현될 때, 이 전체 지연을 전송 지연과 저장 지연으로 배분하는 문제가 있다.

셋째, 네트웍 통로와 종단간의 자원을 동시에 예약하는 알맞은 프로토콜을 개발해야 할 것이다. RSVP 를 사용하는 경우에 라우터들이 저장 매체의 정보도 관리하는 문제가 있어 사용하기 어렵다.

마지막으로, 호 수락을 위해 각 저장 매체의 자원 관리와 네트웍 통로 관리등을 위한 프로토콜을 개발해야 한다. DiffServ 경우에 자원 관리자가 모든 자원을 관리하는데, 이러한 자원 관리에 관한 연구가 많이 진척되지 않았다.

5. 결론

대용량 저장 매체를 필요로 하는 응용 프로그램들이 출현하고, 네트웍에는 수 많은 replica 가 설치되었다. 이러한 환경에서 DNS 는 응용 프로그램의 QoS 에 맞추어 네트웍 통로와 인접한 저장 매체를 할당하는 통합적인 스토리지 QoS 를 제공한다. 그러나, 스토리지 QoS 분야는 태동 단계로써 많은 실용화를 위해서는 많은 연구가 필요한 단계이다. 본 논문은 기존의 연구 방향을 요약 정리하고 향후 연구 문제들을 분석한 것에 의의가 있다.

참고문헌

[1] "iSCSI" - Julian Satran, IETF Internet Draft IP Storage, 24-Jan-03

[2] "Fibre Channel over TCP/IP" - M. Rajagopal, E. Rodriquez, R. weber, IETF Internet Draft IP Storage

[3] "Introducing HyperSCSI" - Modular Connected Storage Architecture Group, Network Storage Technology Division

[4] "HyperSCSI Protocol Specifications" - Modular Connected Storage Architecture Group, Network Storage Technology Division

[5] "The Internet Backplane Protocol: Storage in the Network" - James S. Plank, Micah Beck, NetStore99: The Network Storage Symposium, Seattle, WA, USA, 1999

[6] "Distributed Network Storage Service with Quality of Service Guarantees" - John Chung-I Chuang, Marvin A. Sirbu, Journal of Network and Computer Applications 23(3): 163-185, July 2000. Also in Proceedings of the Internet Society INET'99 Conference, San Jose CA, June 22-25 1999

[7] "End-to-End Quality of Service for High-End Applications" - Ian Foster, Alain Roy, Volker Sander, Linda Winkler, Accepted to Computer Communications, Special Issue on Network Support for Grid Computing 2002

[8] "Storage Resource Managers: Middleware Components for Grid Storage" - Arie Shoshani, Alex Sim, Junmin Gu, Nineteenth IEEE Symposium on Mass Storage Systems, 2002 (MSS '02)

[9] "G-QoSM: A Framework for Quality of Service Management" - R. J. Al-Ali, O. F. Rana, and D. W. Walker, in Proceedings of the UK e-Science Programme All Hands Meeting 2003, held 2-4 September 2003 in Nottingham, UK

[10] "RSVP: A New Resource ReSerVation Protocol", Lixia Zhang, Stephen Deering, Deborah Estin, Scott Shenker, and Daniel Zappala, RFC 2205, September 1997, Proposed Standard

[11] "Resource Allocation for stor-serv: Network Storage Services with QoS Guarantees", John Chung-I Chuang, Proceedings of Internet2 Network Storage Symposium 1999, Seattle WA, October 1999.

[12] "A Network Bandwidth Computation Technique for IP Storage QoS", Youngjin Nam, Junkil Ryu, Chanik Park. Korea Information Science Society 2004, Gwang-Ju