

SNMP 기반의 Web100 라이브러리를 이용한 그리드 응용 성능 측정 방법에 관한 연구

윤민우*, 김동수*, 변옥환**, 정태명*

*성균관대학교 컴퓨터공학과

** 한국과학기술정보연구원

e-mail : mwoon@imtl.skku.ac.kr

A Study on Measurement Methodologies for GRID Application-specific Network Characteristics by using Web100

Min-Woo Yoon*, Dong-Soo Kim, Ok-Hwan Byeon**, Tae Myung Jung*

*Dept. of Computer Engineering, SungKyunKwan University

** Korea Institute of Science and Technology Information

요 약

그리드는 네트워크를 통한 대용량 데이터의 저장과 전송, 그리고 지리적으로 분산된 컴퓨팅 자원의 통합된 활용을 가능하게 해주는 차세대 분산 컴퓨팅 환경이다. 그리드 응용은 이러한 고성능 네트워크 환경에 부합하는 최적화 된 성능 구조가 요구되는데, 특히 초기 그리드 환경에서 응용의 성능 문제를 정확하게 진단하여 산출된 성능 정보는 그리드 응용 자체의 문제점을 찾아 디버깅하거나, 네트워크의 문제점을 발견하여 개선하는 기초 자료로 활용될 수 있다. 본 논문은 그리드 응용의 성능 측정 및 분석 방법에 대한 명확한 요구사항을 제시하고, 요구사항에 부합하는 해결 방법과 기술적 이슈들을 제시한다.

1. 서론

그리드 기술 개발을 주도하고 있는 GGF(Global Grid Forum)는 GGF4 에서 그리드와 웹 서비스 기술을 통합한 OGSA(Open Grid Service Architecture) 모델을 발표했다. 이를 기반으로 최근 발표된 글로버스 툴킷의 새로운 버전인 GT3 는 GT2 가 제공하는 서비스와 개방형 환경에 적합한 프로토콜 suite 및 그리드 서비스 개발을 위한 확장 도구들을 제공한다. 개방형 그리드 모델의 출현은 그리드 기술 전개에 보편성과 타당성의 가치를 부여하며, 관련 응용 개발 의지를 성숙 시키고 있다 [1]. 그리드 응용은 고성능 네트워크 기반구조 상에서 대용량 데이터의 처리 및 분산 컴퓨팅 자원의 효율적인 통합 활용, 멀티캐스팅, 멀티미디어 협업 환경의 구현 등과 같이 기존 네트워크 응용들이 수행할 수 없었던 고성능, 고효율 구조를 요구하고 있다. 그리드 응용의 성능 문제 해결은 초기 개발 이후 실제 네트워크 환경에서 실행 간에 측정되는 여러 가지 유

용한 진단 데이터의 분석을 기초로 지속적인 성능 개선 작업과 디버깅이 요구된다. 따라서 정확한 응용 성능 측정 도구의 개발과 측정된 데이터를 기초로 가공된 유용한 진단 정보가 응용 개발자에게 응용의 실행과 더불어 실시간으로 제공되어야만 한다.

일반적으로 네트워크 기반 응용의 성능 측정 도구는 측정 데이터에 대한 신뢰성과 도구 자체의 시스템 자원 잠식률 정도를 통해 도구 자체의 성능을 입증할 수 있다. 이를 기초로 그리드 응용 성능 측정 도구 설계 시에는 다음과 같은 사항이 고려될 수 있다:

- ① 도구에 의한 호스트 성능 및 네트워크 성능 잠식 최소화,
- ② 도구에 의한 성능 분석 데이터의 유효성 및 신뢰성 극대화.

논문을 위의 고려사항을 바탕으로 SNMP 기반의 Web100 라이브러리 및 NMWG 에서 분석하고 있는

다양한 그리드 네트워크 측정 인자들을 사용하여 그리드 응용의 성능 측정 방법에 대한 솔루션을 제안하고 도구 자체의 성능 향상 방안을 논하며 이와 관련된 기술적 이슈들을 제시할 것이다. 2 장에서는 관련 연구를 설명하고, 3 장에서는 위에서 언급하고 있는 그리드 응용 성능 측정 시 고려사항을 설명한다. 4 장은 고려사항에 따르는 측정 도구의 성능 향상 방안과 기술적 이슈를 살펴볼 것이다. 5 장은 결론 및 향후 연구 과제를 기술한다.

2. 관련 연구

2.1 Web100 프로젝트와 SNMPv2 프레임워크

고성능 네트워크 상에서 응용이 낮은 성능으로 수행된다면, 그 이유는 두 가지 요인으로 분석될 수 있는데, 첫 째는 TCP 모듈과 같은 호스트의 시스템 소프트웨어가 낮은 대역폭 환경에 알맞도록 최적화 되어 있기 때문이고, 둘째는 종단 호스트에서 성능 문제를 진단하기 위한 효과적인 설비나 도구가 부족하기 때문이다. Web100 솔루션은 종단 호스트의 TCP 성능을 정확하게 측정 및 진단하고, 실제 시스템 소프트웨어의 네트워크 패킷 전송 모듈을 수정 함으로써 고성능 네트워크에 최적화된 응용 성능을 구현할 수 있게 한다. Web100 프로젝트는 이와 같은 일련의 프레임워크가 다양한 분야에 적용될 수 있도록, 리눅스 커널의 수정 모듈을 공개하고, 그를 통해 커널 레벨의 /proc 파일 시스템부터 추출되는 다양한 TCP 정보를 SNMPv2 기반의 표준 MIB 으로 정의한 드래프트 문서를 발표하고 있다. 또한, 해당 MIB 의 실제 구현 및 접근 메커니즘을 위한 SNMP 에이전트 구현 라이브러리를 제공한다 [2][3]. Web100 은 SNMPv2 Framework 를 기반으로 에이전트의 구현과, MIB 을 정의한다. 인터넷의 표준 네트워크 관리 프로토콜인 SNMP 는 데이터에 독립적인 통신 메커니즘과 통신 메커니즘에 독립적인 데이터 정의를 통해 모듈화된 개발 프레임워크를 지원한다. SNMPv2 는 bulk 데이터 전송 및 관리 호스트간의 통신 기능을 제공하고 있다 [4].

2.3 Iperf 와 Gloperf

기존에 사용되고 있는 대표적인 그리드 성능 측정 도구에는 Iperf 와 Gloperf 가 있다. 이번 연구에서 그리드 응용의 대표적인 성능으로서 고려되는 다양한 대역폭 측정에 대해 우리는 Iperf 쿼드 모드 메커니즘의 도입을 고려하고 있다. Iperf 쿼드 모드는 TCP 알고리즘의 slow-start 기간을 측정 시간에서 배제 함으로써 측정도구의 정확성과 측정 시간의 효율성을 극대화 한다. 하지만, Iperf 의 측정 인자는 Achievable bandwidth 로 제한되며, 그 외의 loss, delay 측정도 그리드 응용의 성능 인자에 대한 이해가 고려되지 않았기 때문에 그리드 응용 성능 측정 데이터로서 효용성이 없다[5][6].

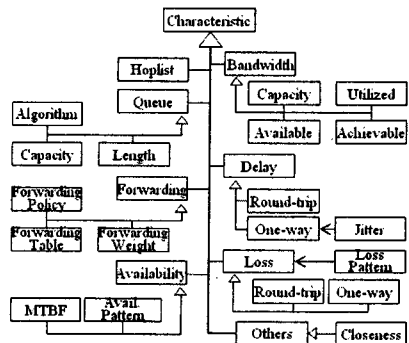
그리드 컴퓨팅 환경에서 네트워크 성능을 모니터링 하기 위한 도구는 그리드 기반구조 자체의 일부분이 되어 할 필요가 있다. 이에 대해 Gloperf 는 훌륭한 인식성을 갖는다. Gloperf 는 그리드 구조의 각 요소와 연동하며 실시간 측정 데이터를 그리드 내부 정보 시

스템(MDS 등)을 통해 사용자에게 제공할 수 있다. 하지만 Gloperf 는 도구 자체의 성능이 네트워크 자원을 과도하게 잠식할 뿐만 아니라, OGSA 환경에서 제안되고 있는 인터넷 관리 구조와 통합 측면에서의 모니터링 개념에 대한 적응력이 떨어진다[7].

이에 따라, 논문에서는 도구의 자원 잠식률을 최소화 하며, 그리드 응용 성능 인자에 대한 높은 이해력을 지닌 정확한 데이터의 산출이 용이하도록 도구 자체의 성능 향상 방안을 도모하고 OGSA 환경의 기존 관리 가능 구조에 통합될 수 있는 SNMP 기반의 네트워크 관리 모니터링 개념의 응용 성능 측정 도구에 대한 요구사항을 도출한다.

3. 그리드 응용 성능 측정 고려사항

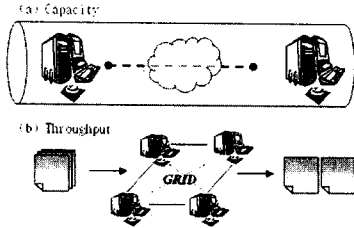
그리드 응용의 성능 인자는 앞서 밝힌 NMWG 의 결과물을 활용할 수 있다. <그림 2>는 기존 측정 인자들을 가지고 NMWG 가 특성화시킨 인자들의 계층적 관계를 보여준다. 추가적으로 제시될 수 있는 사항은 네트워크 응용을 포함한 일반적인 응용의 성능 인자로서 응용 응답 시간(Application Response Time, ART)과 관련된 인자가 있다. 1996 년 HP 와 Tivoli 에 의해서 발표된 벤더 중립적인 개방형 ARM API 는 사용자 관점에서 정의되는 응용과 트랜잭션의 응답 시간 측정 방법을 제공하고 기초적인 성능 모델의 설계를 가능하게 한다 [8].



<그림 1> Hierarchy of network characteristics

3.1 그리드 응용 성능 인자 및 측정 방법

그리드와 같은 분산 컴퓨팅 환경에서 네트워크와 종단 호스트들은 하나의 시스템으로서 접근 가능하다. 즉, 그리드 응용의 성능은 네트워크의 성능과 호스트의 성능, 응용 자체 성능의 종합적인 상호 작용으로 이루어진다. <그림 3> (a)의 원통 그림은 bandwidth 와 RTT 의 곱으로 정의되는 네트워크의 용량(capacity)을 나타낼 때 주로 사용되는 그림인데, 여기서는 하나의 그리드 시스템이 수용할 수 있는 작업량을 표현한다. 실제로 원통으로 표현되는 시스템의 구성요소(Network entity)들은 시스템의 성능에 영향을 미칠 수 있는 다양한 성능 인자들을 내재하는데, NMWG 는 [9]에서 각 구성요소의 상호 연관성을 정의 하고 있다 [9][10].



<그림 3> Grid Application Performance Characteristics

중단간 성능의 관점에서, 단위 프로세스나 스레드의 집합에 의해 하나의 그리드 시스템에서 수행되는 트랜잭션은 용량(capacity)과 단위시간 처리량(throughput)의 응용 응답 시간에 대한 단일 큐 모델로 표현될 수 있는데, 이 모델은 응용 성능의 기준으로 제시될 수 있는 이상적인 성능 기준치를 제공할 수 있다(이 값은 응용의 종류에 의존적이다). 이와 비교될 용량과 처리량의 계산을 위해서 측정되어야 하는 값이 <그림 2>에서 계층적으로 정의된 그리드 응용의 성능 인자들이다. 실제로 용량이나 단위시간 처리량에 대해 성능의 의존 요소는 네트워크일 수도 있고 호스트 플랫폼일 수도 있으며, 해당 응용의 메커니즘 자체 일 수도 있다. 그러므로 응용 진단의 목적은 응용 성능의 의존 요소를 밝히는 일도 될 수 있다. 진단에 사용될 수 있는 예제로서 <표 1>은 <그림 2>에서 대역폭 구성 인자들에 대한 의존 요소를 나타낸다. Dependency 필드의 내용은 확정적인 요소로서 볼 수 없는데, 실제로 Usage bandwidth는 네트워크와 플랫폼의 영향을 포함하고 Achieveable bandwidth 역시 네트워크 영향을 포함하기 때문이다.

<표 1> Dependency of the Bandwidth

Bandwidth	Dependency	Layer	Capacity
Available	Network	Link	× RTT
Achievable	Platform	IP	× RTT
Usage	Application	Application	× ART

** ART: Application Response Time

네트워크 측정은 크게 Active 방식과 Passive 방식으로 나뉜다. <표 1>에서 Available 과 Achievable bandwidth의 경우 직접 probe를 생성하는 Active 방식이 적합하다. 이외에 측정 방식 별로 네트워크 인자를 분류한 것이 <표 2>에 나와있다.

<표 2> Classification-based Measurement Methodology of Network Characteristics

Active-adapted	Passive-adapted
Available/Achievable bandwidth, Delay/Jitter, Loss	Usage bandwidth, Availability (service perspective)

4. 측정 도구 성능 향상 방안 및 기술적 이슈

4.1 측정 도구 성능 향상 방안

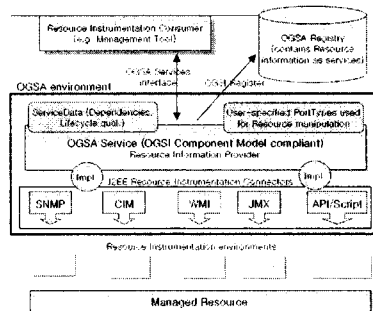
Active 방식의 대역폭 측정은 Iperf 킷모드에서 구현한 메커니즘에서와 같이 TCP의 slow-start 기간 예측에

의해서 정확도 증가 및 측정 속도 향상 등의 도구 자체 성능 향상을 이루어 낼 수 있다. 실제로 slow-start 메커니즘의 특성상 이 기간의 대역폭은 실제 중단간의 획득가능 대역폭의 20%내외가 된다. Web100 프로젝트에서 도출된 slow-start 기간의 계산식은 다음과 같다.

$$Slow\text{-start\ duration} = \lceil \log_2(\text{ideal window size in MSS}) \rceil \times RTT$$

기간 계산을 위해 요구되는 실제 값은 Web100에서 정의한 MIB 과 agent를 통해 획득 가능한데, 내용은 다음과 같다: ①Current congestion window, ②Maximum, minimum and smoothed RTT, ③Receiver advertised window, ④Maximum segment size, ⑤Data bytes out [6].

Web100의 커널 패치는 리눅스(kernel 2.4.21)의 /proc 파일시스템에서 제공하는 TCP 파라미터 값을 통해 해당 정보를 TCP 확장 MIB에 저장하고, Web100 라이브러리는 해당 MIB에 접근할 수 있는 접근 메커니즘을 위한 agent를 구현한다. 실제로 이와 같은 Active 방식의 측정에 있어서 Iperf 킷모드 모듈을 채용하여 대역폭 측정의 성능을 높이는 것은 전체 측정 구조의 일부가 되는데, Active 방식의 측정 인자에 대한 전체적인 측정 성능 향상을 위한 방법으로 동일 링크 상의 원격 호스트를 성능 관리 호스트로 활용하는 방법이 가능하다. 실제로 계산 그리드와 같이 거대한 분산 시스템은 성능 분석이나, 성능 튜닝, 성능 예측, 작업 스케줄링 등과 같은 다양한 관리 작업을 위한 방대한 양의 모니터링 데이터를 필요로 한다. 이에 대해 그리드 자원 모니터링 구조의 하나로써 관리 대상 자원의 정보가 다양한 에이전트 시스템에 의해서 제공될 수 있다. 로컬에서 관리 대상 자원의 상태에 대한 정보를 제공하는 로컬 에이전트 시스템으로서 센서 기반의 그리드 환경 모니터링 시스템이나[11], SNMP와 NetLogger[12]등을 기반으로 하는 다양한 에이전트 시스템이 제안되고 있다. <그림 4>는 OGSA 프레임워크로서 제시되고 있는 관리 가능 구조를 보여준다.



<그림 4>OGSA Manageability Architecture

인터넷 표준 네트워크 관리 프로토콜인 SNMP는 구현이 손쉬울 뿐 아니라, 단순 명료한 동작특성을 갖는다. <그림 4>에서와 같이 OGSA 환경의 일부 모듈로서 SNMP의 지원 방안이 제시/구현됨에 따라, 로컬

에이전트 방식의 응용 성능 측정의 구현 가능성은 높다. GGF Performance WG 에서 로컬 에이전트 방식의 모니터링은 효과적인 성능측정을 가능하게 할 뿐 아니라, 응용의 성능에 특화된 다양한 인자를 정책적으로 수집할 수 있다. Web100 라이브러리를 통해, 리눅스 커널상의 다양한 호스트 정보와 특정 응용의 호스트 대비, 혹은 네트워크 대비 성능정보를 그대로 모니터링 데이터에 조합할 수 있으므로 최종적으로 신뢰성과 유용성을 갖춘 응용의 성능 진단 데이터를 도출해 낼 수 있다.

4.2 기술적 이슈

Web100 기반의 그리드 응용 성능 측정 시스템의 개발에는 다음과 같은 두 가지 기술적인 이슈들이 존재한다.

Platform Dependency: NLNLR 의 고성능 네트워크 성능 관련 프로젝트 일환으로 진행되었던 Web100 프로젝트는 현재 기존 펀딩(funding)이 중단되었고, 최종적인 결과물로서 제공되고 있는 Web100 라이브러리는 Linux kernel 2.4.21 버전상에서 구현되어있으며, 이것은 다른 OS 와 호환되지 않는다. 이러한 이식성 문제는 OGSA 의 개념과 상반되고, 실제로 거대한 분산 시스템에서 특정 플랫폼에 종속적인 라이브러리는 사용될 수 없다. 특히 라이브러리가 접근하고자 하는 TCP 확장 MIB 이 여전히 드래프트 상태이고, 해당 MIB 의 구현과 에이전트가 각 벤더에서 능동적으로 구현되고 지원되는 것은 요원한 상태이다. 현재로서는 자발적인 개발자 그룹을 통해 SNMP 를 통한 가능한 모든 플랫폼에 적용될 수 있는 일관성 있는 인터페이스가 개발되는 것을 계획할 수 있다.

Security: Community 메커니즘을 기반으로 하는 SNMPv2 의 보안 문제점은 각 GRID 사이트의 독자적인 관리 정책에 위배됨으로 실제로 전체 분산시스템의 모니터링에 적용되기 힘들다. GRID 환경의 보안 정책과 SNMP 프로토콜을 사용함으로써 발생할 수 있는 보안 문제점이 상충되는 가장 두드러진 부분은 Community 기반의 SNMP 보안 프로토콜이 GRID 의 자원의 중요한 정보들을 외부에 유출 시키는데 악용될 수 있다는 점이다. 이에 대한 해결책은 두 가지 접근 방법이 고려될 수 있다. 첫 번째는, SNMP 의 사용자에게 인증되어 사용될 수 있는데, 사용자 기반의 보안 메커니즘을 사용하는 SNMPv3 의 사용과, IPsec 등의 추가적인 보안 기술 적용이 요구된다. 두 번째 접근은 실제 OGSA 환경의 보안 메커니즘상에 SNMP 프로토콜을 흡수하는 것인데, SNMP 프로토콜의 부분적 수정이 요구됨으로 표준화에 위반될 수 있다. 하지만 역시 GRID 의 'single sign on'과 연동되는 일련의 인증 메커니즘이 새로운 SNMP 프레임워크로 제시될 수 도 있다.

5. 결론 및 향후 계획

분산 시스템의 응용을 진단 함으로서 네트워크와 호스트 플랫폼, 응용 자체의 문제점을 찾아내고 개선

할 수 있다. 응용의 성능을 측정하는 것은 진단의 중요한 기초 자료를 제공하는데, 전체적인 진단 시스템의 일부로 SNMP 프레임워크가 제공될 수 있다. 미국의 고성능 네트워크의 TCP 성능 진단 및 향상에 대한 연구로서 Web100 프로젝트는 인터넷 표준인 SNMP 를 이용한 모델의 기초와 일부 구현 물을 제공하고 있다. 이를 바탕으로 본 논문은 GRID 응용의 성능관련 프레임워크와 그에 따르는 진단 도구의 성능 향상을 위한 다양한 고려사항을 살펴보았다.

본 논문에서 분석된 자료를 기초로 OGSA 의 자원 모니터링과 그리드 관리 모델 및 초기 그리드 환경의 문제점 진단을 통합하는 거시적인 관점에서 SNMP 구조 적용을 위한 프레임워크에 대한 연구를 진행할 계획이다.

참고문헌

- [1] I. Foster, D. Gannon, "The Open Grid Service Architecture Platform", draft-ggf-ogsa-platform2, 2003. 2. 16.,
- [2] Matt Mathis and three others, "TCP Extended Statistics MIB", Internet-Draft, 2002.11.3.,
- [3] Web100 Site, "<http://www.web100.org>",
- [4] J. Case and four others, "Introduction to Version 3 of Internet Standard Management Framework", RFC 2570, 1999. 4.,
- [5] Mark Gates, Ajay Tirumala, Jon Dugan, Kevin Gibbs, "Iperf User Docs", NLNLR application support, March 2003, http://dast.nlnlr.net/Projects/Iperf/iperfdocs_1.7.0.html.
- [6] Ajay Tirumala, Les Cottrell, Tom Dunigan, "Measuring end-to-end bandwidth with Iperf using Web100", PAM2003, 2003. 4.,
- [7] Craig A.Lee, James Stepanek, Carl Kesselman, Rich Wolski, Ian Foster, "A Network Performance Tool Grid Environments", Proc. of Supercomputing '99 (Portland, OR. Nov. 13, 1999), pp. 1-16.
- [8] Yiping Ding, "Performance Modeling with Application Response Measurement (ARM): Pros and Cons", www.bmc.com/offers/performance/whitepapers/docs/1997/performance_modeling_with_arm.pdf, 1997.,
- [9] Bruce Lowekamp and five others, "A Hierarchy of Network Performance Characteristics for Grid Applications and Services", NMWG in GGF ISP, 2003. 6. 19.,
- [10] NMWG Site, "<http://forge.gridforum.org/projects/nmwg/>",
- [11] Brian Tierney and five others, "A Monitoring Sensor Management System for Grid Environments", In Proc. 9th IEEE Symp. on High Performance Distributed Computing, pages 97--104, 2000. 6.,
- [12] Tierney, B. and five others, "The NetLogger Methodology for High Performance Distributed Systems Performance Analysis", Proceeding of IEEE High Performance Distributed Computing conference, July 1998, LBNL-42611, <http://www.didc.lbl.gov/NetLogger/>