

## 액티브네트워크상의 웹 캐싱을 위한 서비스 컴포지션에 관한 연구

홍성준\* 이 용수\*\*

### A study on service composition for web caching on active network

Sungjune Hong \* Youngsu Lee \*\*

#### 요 약

본 논문은 액티브네트워크 상의 웹 캐싱을 위한 서비스 컴포지션에 관하여 언급하였다. 액티브네트워크에서는 사용자의 요구사항에 맞게 서비스들을 빠르게 재구성하기 위한 서비스 컴포지션에 대한 요구사항이 대두되고 있다. 이러한 요구사항을 지원하기 위해서 우리는 액티브 네트워크상에서 웹 캐싱을 위한 서비스 컴포지션의 설계 및 구현에 관하여 언급하였다.

#### Abstract

This paper describes an application level composition mechanism called Generic Modeling Environment(GME) for web caching on an Application Level Active Network(ALAN). Web caching on an ALAN requires the application level composition mechanism and a service composition to support adaptability for self-organization. ALAN was developed to solve the problems of the network level Active Network(AN). ALAN has the features of both AN as well as mobile agents. The efficient composition mechanism for the existing AN projects has been supported primarily for the network level AN. Conversely, ALAN lacks support for the application level AN. The existing web caching technology is inter-connected in a manually configured hierarchical tree. Since a self-organization system is intended to be adaptive, web caching for self-organization does not involve a manual configuration or any low-level tuning of the individual nodes of the entire system but requires service composition to support adapting intelligence and fault-tolerance to enable self-organization.

---

\* 여주대학 정보통신과 조교수

\*\* 여주대학 컴퓨터 정보관리과 부교수

## I. 서론

최근 컴퓨터 시스템은 단순하고 중앙 집중적인 구조에서 분산 컴퓨팅을 지원하는 복잡한 형태로 진화되어왔다. 액티브네트워크(Active Network)는 이런 현재의 수동적인 네트워크의 진화라고 볼 수 있다. 액티브네트워크의 장점은 네트워크를 프로그래머블하게 함으로 긴 표준화와 배치(deployment)시간을 줄일 수 있다는 점이다. 액티브네트워크(1)(2)(3)(4)(5)는 ANTS, CANE등의 프로젝트에서 주로 네트워크 라우터 수준에서 진행되어 왔다. 그러나 이러한 시도는 비현실적이다. 왜냐하면 네트워크 제공자가 네트워크 수준에서 제 3자로부터 프로토콜 코드의 배치를 허용하려고 하지 않기 때문이다.

이런 문제점을 해결하기 위해서 최근에 ALAN(6)이라는 응용수준에서의 액티브네트워크가 등장하였다. ALAN의 장점은 액티브네트워크와 이동 에이전트의 장점을 가지면서 현재 인터넷에 적용하기에 쉽도록 되어 있다.

최근에는 CCML(Component Compatibility Markup Language)(7)와 같은 ALAN과 관련된 연구가 진행 중에 있지만 아직 ALAN은 서비스 컴포지션에 대한 고려가 부족하다.

또한, 웹이 점점 성장하고 복잡해짐에 따라 웹 캐싱(9)(10)(11)(12) 또한 같이 복잡해져 가고 있다. 이러한 복잡성 문제를 해결하기 위한 방법으로 서비스 컴포지션 방법이 요구되고 있다. 서비스 컴포지션 방법에는 OMG(Object Management Group)의 MDA(Model Driven Architecture)와 GME(Generic Modeling Environment)(8)가 있다. MDA는 주로 비즈니스 응용에 사용되고 GME는 실시간과 임베디드 시스템에 응용된다.

이러한 요구사항을 지원하기 위해서 본 논문에서는 액티브 네트워크상에서 웹 캐싱을 위한 서비스 컴포지션을 제안하였다.

본 논문에서는 컴포지션 메카니즘으로 MIC(Model - Integrated Computing)기반의 GME(Generic Modeling Environment)를 사용하였다. 여기서 MIC는 모델링 언어를 시스템의 실제 코드에 적용하는 기술을 의미하고

GME는 MIC기술을 구현하는 개발 도구를 의미한다.

GME 지능망에서 유래된 SCE(Service Creation Environment) 유사한 메카니즘을 가지고 있다. 2장에서는 ALAN상에서 서비스 컴포지션을 이용한 웹 캐싱에 관하여 언급하였고 3장에서 본 웹 캐싱의 구현에 관하여 언급하였다. 4장에서는 본 웹 캐싱의 수행결과 그리고 5장에서 결론을 맺었다.

## II. 관련 연구

### 1. ALAN

액티브네트워크 연구는 주로 네트워크 라우터 수준에서 개발 진행되고 있다. 그러나 영국 UCL(University College London)에서 제안된 ALAN(Application Level Active Network)은 네트워크 수준에서 문제점을 보완하기 위해서 응용 수준의 액티브네트워크를 제안하였다. ALAN은 웹 과 같은 클라이언트와 서버로 구성되어 있고 서버와 클라이언트사이의 통신은 EEP(Execution Environment for Proxylet)와 프로그래머블하고 다운로드 가능한 프락시릿(Proxylet)에 의해서 성능 향상된다. 현재의 EEP는 웹 캐싱 하부구조를 가지고 있고, EEP는 브라우저와 웹 서버 사이에서 가장 가까운 최적의 거리에 있는 캐시 데이터를 찾아서 연결시키는 기능을 한다. EEP와 통신은 RPC를 통해서 가능하다. 현재 구현이 자바이기 때문에 RMI(Remote Method Invocation)가 사용된다. EEP상에서는 프락시릿이 동작한다. 여기서 프락시릿은 EEP에서 다운로드되고 동작하는 동적 코드를 의미한다. 일반적으로 ALAN은 네트워크의 전략적 위치에 있는 수많은 EEP로 구성된다.

### 2. GME

GME는 도메인 기반 모델링과 프로그램 종합 환경(Program Synthesis Environment)를 지원하는 구성 가능한 도구를 의미한다. GME는 MIC를 기반으로 하는 기술로 MIC는 도메인 모델을 실제 구현 코드로 자동으로 변환시켜준다. GME는 넓은 범위의 도메인을 구성가능한 CDSDE(Configurable Domain Specific Development

Environment)에 기초한다. GME 2000은 도메인 기반 설계환경을 만들기 위한 개발 도구이다. GME 2000은 메타모델을 이용하는데 메타모델은 UML과 OCL(Object Constraint Language)로 구성되어 있다. GME2000의 메타 모델은 메타 모델 컴포지션을 지원할 수 있다. GME 2000의 특징은 메타모델 컴포지션을 지원하기 위한 메타 모델 환경을 지원한다는 점이다.

### III. 웹 캐싱을 위한 서비스 컴포지션의 설계

#### 1. 웹 캐싱을 위한 서비스 컴포지션의 구조

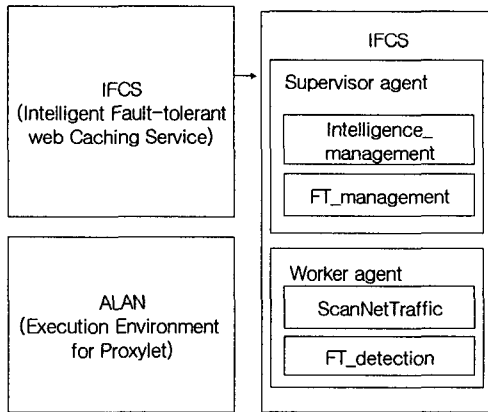


그림 1. 웹 캐싱을 위한 서비스 컴포지션의 구조

〈그림 1〉은 웹 캐싱을 위한 서비스 컴포지션의 구조를 보이고 있다. 본 논문에서는 웹 캐싱을 위한 컴포지션으로 지능성과 결합허용성을 웹 캐싱에 적용하는 서비스를 설계하였다.

본 논문에서는 이 서비스를 IFCS(Intelligent Fault-tolerant web Caching Service)라 명칭하였다.

웹 캐싱을 위한 서비스 컴포지션은 Intelligent\_management 모듈, FT\_management 모듈, Multicast\_management 모듈, Replication\_Management 모듈 등을 포함한다. Supervisor는 각 모듈을 관리하고 각 모듈 내부에는 Worker모듈이 포함되어 있다.

#### 2. 웹캐싱의 동작

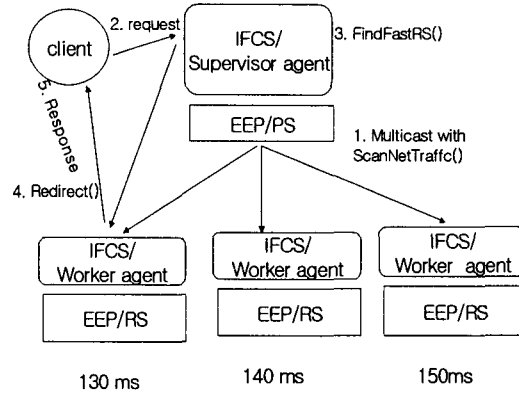


그림 2 웹 캐싱의 동작 구조

〈그림 2〉는 클라이언트에게 가장 빠른 RS를 찾아서 연결시키기 위한 웹 캐싱 동작을 보이고 있다.

- ① Supervisor는 multicast\_management 모듈에 의해서 멀티캐스트 메시지를 보내고 RS로 파견된 Worker로부터 대역폭과 같은 네트워크 상태의 정보를 모을 수 있다.
- ② 클라이언트가 PS상에 데이터를 요청할 때 IFCS는 Worker로부터 얻은 네트워크 상태 정보로부터 가장 빠른 RS를 발견할 수 있다.
- ③ IFCS는 Intelligent\_management 모듈에 의해서 PS로 연결시키는 대신에 가장 빠른 RS로 우회(redirect)시킨다.
- ④ 클라이언트는 가장 빠른 RS로부터 데이터를 얻는다.

### IV. 웹 캐싱을 위한 서비스 컴포지션의 구현

#### 1. GME 도구를 사용하는 웹 캐싱을 위한 서비스 컴포지션

본 논문은 컴포지션 메카니즘으로 GME를 사용한다. GME는 응용 수준에서 MIC와 ALAN을 통합하는데 사용

된다. 그림 3은 GME도구에 의해서 컴포넌트 기반 시스템의 컴포지션을 보인다. 컴포지션 다이어그램은 각 컴포넌트간의 관계를 보이고 있는데 크게 Client, Supervisor, Worker로 구성되어 있고 Supervisor와 Worker의 컴포지션 다이어그램은 Supervisor와 Worker의 각 내부에 정의되어 있는 모듈들간의 관계를 보이고 있다.

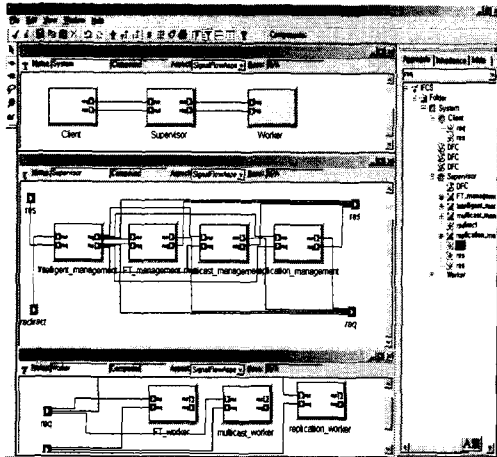


그림 3. GME도구에 의한 웹 캐싱을 위한 컴포지션

## 2. ALAN상에서 웹 캐싱을 위한 서비스 컴포지션에 따른 구현 코드

```

public IFCS implements Proxylet {
    Public IFCS() { }
    public Object Run( String parameter){
        String url = new String("PS");
        IFWS IFCSRef = (IFCS) Namaing.lookup(url);
        String intelligent = IFCSRef.supervisor();
        return "OK"
    }
}
    
```

그림 4. 웹 캐싱을 위한 서비스 컴포지션에 따른 구현 코드

(그림 4)는 ALAN상에서 웹 캐싱을 위한 서비스 컴포지션에 따른 구현 코드를 보이고 있다. ALAN상에서 구현 코드는 Proxylet이라는 기본 클래스에서 상속되어 구현된다. Supervisor모듈은 intelligent\_management 모듈, FT\_management 모듈, multicast\_management 모듈, replication\_management 모듈을 통제한다.

## V. 웹 캐싱 실행 결과

### 1. 지능성 테스트

(그림 5)는 가장 빠른 RS를 선택하는 실험을 보이고 있다. 수행 결과는 각 RS의 대역폭을 보이고 그 중에서 가장 빠른 RS를 선택하는 과정을 보이고 있다

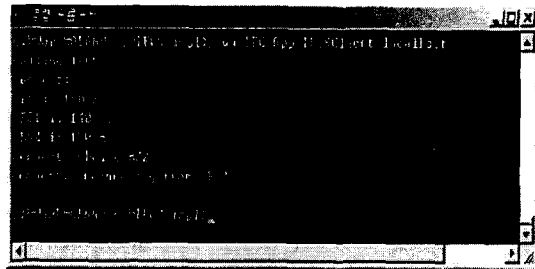


그림 5. 가장 빠른 RS를 선택하는 실험

### 2. 결함 허용성 테스트

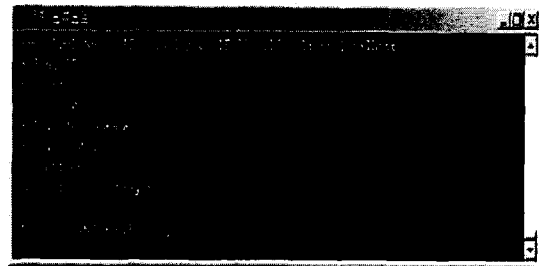


그림 6. 결함 허용성 실험

(그림 6)은 웹 캐싱의 결함 허용의 실험 결과를 보이고 있다. RS중에 하나가 시스템 결함이 발생했을 때 그 RS는 제공하지 못하기 때문에 그 RS를 제외한 RS중에서 가장 빠른 RS를 찾는 과정을 보이고 있다.

## VI. 특성 비교 및 성능 평가

### 1. 특성 비교

표1. 주된 특성의 비교

	AWC	ALAN상의 웹 캐싱	본 논문의 웹 캐싱
제어신호 트래픽 감소	X	O	O
데이터 트래픽 감소	X	O	O
지능성	X	O	O
결합허용	X	X	O
적응성	X	X	O

〈표 1〉은 AWC[11]와 기존 ALAN의 웹 캐싱 그리고 본 논문의 웹 캐싱의 주된 특성의 비교를 보이고 있다.

본 논문의 웹 캐싱이 기존의 ALAN의 웹 캐싱보다 결합 허용성과 적응성을 더 지원하는 것을 보인다.

### 2. 성능 평가

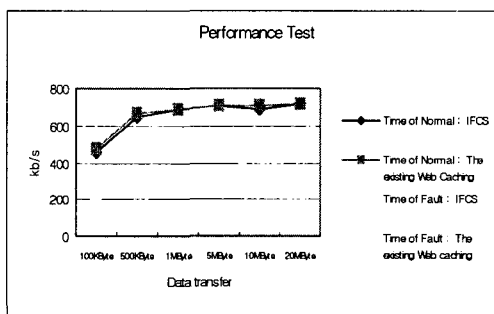


그림 7. 성능 평가

〈그림 7〉은 성능평가의 결과를 보이고 있다. 기존 ALAN상의 웹 캐싱과 본 논문의 웹 캐싱을 비교하면 네트워크가 정상적인 경우에는 차이가 없다. 그러나 장애가 발생하는 경우에 본 논문의 웹 캐싱이 성능이 향상되는 것을 보인다.

## VI. 결론 및 향후 연구

본 논문은 웹 캐싱을 위한 서비스 컴포지션 방법을 제시하였다. 향후 연구로는 컴포지션을 이용한 차치 적응성에 대한 연구가 필요하고 결합허용 발견과 복구에 대한 연구가 필요하다.

### 참고문헌

- [1] Michael Hicks, Pankaj Kakkar, Jonathan T. Moore, Carl A. Cunter, and Scott Nettles, "PLAN: A packet language for active networks," in proceedings of the 1998 ACM SIG-PLAN International Conference on Functional Programming (ICFP'98), September 1998.
- [2] Tennenhouse D and Wetheral DS, "Toward an active network architecture", computer communication review, 26, No 2, 1996.
- [3] S. Bhattacharjee, K. Calvert, Y. Chae, S. Merugu, M. Sanders, and E. Zegura, "CANEs : An Execution Environment for Composable Services" in Proceedings of DANCE 2002.
- [4] S. da Silva, D. Florissi and Y. Yemini, "Composing Active Services in NetScript," DARPA Active Networks Workshop, Tucson, AZ, March 9-10, 1998, in Netscript, <http://www.cs.columbia.edu/dcc/netscript/Publications/publications.html>.
- [5] AN Composable Services Working Group. Composable Services for Active Networks, Sept. 1998.

- <http://www.cc.gatech.edu/projects/canes/papers/cs-draft0-3.pdf>.
- [6] I.W.Marshall, et, al, "Application-level Programmable Network Environment," BT Technology Journal, Vol. 17, No2 April 1999.
- [7] S.Simpson, P.Smith, M.Banfield, and D. Hutchison, "Component Compatibility for Heterogeneous Active Networking", Presented at IEE Informatics, Nov.2000,London.  
<http://www.activenet.lancs.ac.uk/papers/ieeelan2000.pdf>.
- [8] A. Ledeczi et al., "The Generic Modeling Environment," Proceedings of WISP 2001, May, 2001.
- [9] SQUID homepage.  
<http://www.squid-cache.org/>.
- [10] Floyd, S., "Adaptive Web Caching," Boulder Cache Workshop '97, (June 9-10, 1997) in AWC,  
<http://www-nrg.ee.lbl.gov/floyd/web.html>.
- [11] Oliver Aubert and Antone Beugard, "Toward a fine-grained apdaptivity in web caches", in proceedings of the 4th international web caching workshop, April 1999,  
<http://www.ircache.net/workshop/papers/aubert-0.ps.gz>.
- [12] M.Baentsch,A.Lauer, L.Baum,G.Molter., S. Rothkugel, P.Sturm, "Enhanceing the Web's Infrastructure-From Caching to Replication," IEEE Internet Computing, Vol.1,No. 2, March-April 1997.

## 저 자 소 개

### 홍 성 준

1991 경원대학교 전자계산학과 졸업(공학사)

1993 건국대학교 대학원 전자계산학과 졸업(공학석사)

1998 건국대학교 대학원 컴퓨터공학과 졸업(공학박사)

현재 여주대학 정보통신과 조교수  
<연구분야> 컴퓨터네트워크, 전자상거래

### 이 용 수

1986 명지대학교 전자계산학과 졸업(공학사)

1989 건국대학교 대학원 전자계산학과 졸업(공학석사)

현재 여주대학 컴퓨터정보관리과 부교수

<연구분야> 데이터베이스, 전자상거래