호스트 서버에서의 통합 에이전시

신홍섭º, 오세만
동국대학교 컴퓨터공학과
e-mail : (s4064º, smoh)@dongguk.edu

Integrated Agency on the Host Server

Hongseob Shinº, Seman Oh
Dept. of Computer Engineering, Dongguk University

요 약

휴대성을 가장 큰 장점으로 제공하는 모바일 장치는 모바일 장치를 위한 다양한 컨텐츠의 개발로 인해 관심이 주목되고 있다. 이러한 컨텐츠의 형식으로는 StandAlone 방식과 Network 방식이 있으며 근래에는 대부분의 컨텐츠들이 서버와의 통신을 통해 진행되는 후자를 따르고 있다. Network 방식의 컨텐츠는 서버와의 통신이 필수이기 때문에 별도의 서버와 함께 에이전트를 구현해야 한다. 모바일 장치에 탑재된 다양한 가상 기계와 수많은 컨텐츠들은 각각의 에이전트를 필요로 하며, 서버의 에이전트가 증가함에 따라 에이전트를 별도로 관리해야 하는 등의 문제점들이 발생하기 시작하였다.

본 논문에서는 이러한 문제점들을 해결하기 위한 방안으로서 통합된 형태의 에이전시를 제안하고자 한다. 제안된 에이전시는 모바일 장치의 가상기계와 컨텐츠의 종류에 상관없이 단 하나의 에이전시를 통해 상호간의 통신이 가능하다. 뿐만 아니라 서로 다른 모바일 장치와의 상호운용도 가능한 장점을 지닌다. 통합 에이전시는 접속 모듈과 함수 호출 모듈로 구성된다. 접속 모듈은 모바일 장치로부터 전송되는 전송정보 가운데 필요한 정보들을 추출하는 역할을 담당하며, 함수 호출 모듈은 접속 모듈에서 추출한 정보들을 이용하여 모바일 장치가 호출한 서버의 함수를 호출하는 역할을 담당한다.

1. 서론

모바일 장치와 통신기술의 발전은 국내외의 모바일 환경에 커다란 변화를 가져왔다. 셀룰러 폰으로 대표되는 모바일 장치는 기본적인 통신 수단 이외에도 게임, 모바일 결재[6] 등과 같은 컨텐츠를 실행시키는 수단으로 사용되고 있다. 휴대성을 가장 큰 장점으로 제공하는 모바일 장치를 위한 다양한 컨텐츠가 개발될 수 있었던 이유는 모바일 장치에 다양한 가상기계가 탑재되었기 때문이다[1,5]. 모바일 장치에 탑재된 다양한 프로세스 가상기계(GNEX, KVM, WIPI 등)는 모바일 장치를 위한 컨텐츠의 실행 환경으로서 통신사마다 각기 다른 가상기계를 채택하여 사용하고 있으며, 근래에는 WIPI(Wireless Internet Platform for Interoperability)로 통합되는 추세이다.

모바일 장치를 위한 컨텐츠는 Stand Alone 방식과 Network 방식으로 분류할 수 있다. 전자는 별도의 서버 접속 없이 모바일 장치를 이용하여 독립적으로 진행되는 방식의 컨텐츠 형태를 말하며, 후자는 서버와의 통신을 통하여 진행되는 컨텐츠 형식을 말한다[5].

Network 방식의 컨텐츠는 서버와의 통신을 통해 진행되기 때문에 별도의 호스트 서버를 구성해야 한다. 호스트 서버는 Network 방식의 컨텐츠가 접속하는 서버로서 모바일 장치에 탑재된 가상기계의 형식과 컨텐츠의 종류에 따라서 많은 수의 에이전트를 필요로 한다[1]. 호스트 서버에 탑재되는 에이전트의 개수가 증가함에 따라 별도로 에이전트를 관리해야 하는 문제점과 서로 다른 가상 기계 컨텐츠들 사이에서의 정보 교환이 불가능한 문제점들이 발생하였다.

본 논문에서는 이러한 문제점들을 해결하기 위한 방안으로서 여러 에이전트를 통합한 형태의 에이전시를 제시하고자 한다. 통합된 에이전시는 가상기계의 종류와 컨텐츠에 상관없이 단 하나의 에이전시를 통해 모바일 장치의 컨텐츠들과 통신이 가능하다. 또한 서로 다른 가상 기계 컨텐츠들 사이에서도 정보 교환이 가능한 장점을 가진다.

통합 에이전시는 접속 모듈과 함수 호출 모듈로 구성된다. 접속 모듈은 모바일 장치로부터 전송되는 전송정보 가운데 필요한 정보들을 추출하는 역할을 담당하며, 함수 호출 모듈은 접속 모듈에서 추출한 정보들을 이용하여 모바일 장치가 호출한 서버의 함수를 호출하는 역할을 담당한다.

모바일 장치의 컨텐츠가 통합 에이전시와 함께 운용되기 위해서는 별도의 모바일 에이전트가 요구된다. 이러한 모바일 에이전트는 정보 생성기를 통해 손쉽게 작성할 수 있으며, XML 형식의 전송정보를 사용한다. 정보 생성기에 관한 연구는 본 논문의 선행 연구로 진행되었다[5].

본 논문의 2 장에서는 모바일 환경의 가상기계, 에이전트 그리고 정보 생성기에 대해 기술하였으며, 3 장에서는 통합 에이전시의 전체 구성과 각각의 모듈 그리고 세부 컴포넌트에 대해 살펴본다. 그리고 4 장인 결론 및 향후 연구과제에서는 모바일 환경에서 통합 에이전시를 운용함에 따른 장점들과 새로운 가상 기계에 대한 통합 에이전시의 적용기법과 자동화기법에 필요한 향후 연구과제에 대하여 기술하였다.

2. 관련연구

2.1 모바일 장치의 가상 기계

모바일 장치를 위한 프로세스 가상 기계는 가상 기계에서 실행되는 컨텐츠의 개발 언어에 따라 C 언어 계열과 JAVA 언어 계열로 분류할 수 있다.

C 언어 계열의 대표적인 가상 기계로는 신지소프트에서 개발한 GNEX/GVM 을 예로 들 수 있으며, JAVA 언어 계열의 대표적인 가상 기계로는 선 마이크로시스템즈사에서 개발한 KVM 을 예로 들 수 있다. 특히, KVM 의 경우는 국내의 통신사에서 그 형식을 변경하여 사용되고 있으며 모바일 장치를 위한 통합 플랫폼으로 알려진 WIPI 또한 KVM 을 기본으로 제작되었다.

이러한 가상 기계의 등장은 모바일 장치를 위한 컨텐츠를 보다 손쉽게 작성할 수 있는 역할을 담당하였다. 가상 기계가 가지는 플랫폼 독립적인 특징 때문에 각기 다른 하드웨어와 사용자 인터페이스를 가지는 모바일 장치라 할지라도 동일한 형태의 컨텐츠 제작이 가능하다[2,5,6].

2.2 에이전트

에이전트의 정의는 에이전트를 보는 시각에 따라 다양하게 정의되고 있다. 대표적인 정의를 보면, 에이전트는 특정 목적을 수행하기 위하여 사용자를 대신하여 작업을 수행하는 자율적인 프로세스이다. 에이전트는 수동적으로 주어진 작업만을 수행하는 것이 아니고, 자신의 목적을 가지고 그 목적 달성을 추구하는 능동적인 자세를 가진다. 또한, 에이전트는 사람 또는 조직의 권한을 대신하여 독립적으로 수행될 수 있고, 다른 에이전트와 상호 교류할 수 있으며 사용자가 수행 할 작업이나 반복적인 일들을 자동적으로 처리해 주는 프로그램이다. 에이전트는 지시된 작업을 수행하고 저장하기 위한 기능을 제공하는 제어지식 에이전트와 수행할 역할의 기능을 규정하고 있는 영역지식 에이전트, 그리고 프로시저 호출과 메시지 교환을 처리하는 통신 모듈 에이전트로 구성된다[1,7]. 또한 제어지식 에이전트, 영역지식 에이전트, 통신 모듈에이전트와는 별도로 분산처리 시스템의 패러다임에서 시작된 모바일 에이전트의 경우는 네트워크상의 사용자를 대신하여 실행되는 애플리케이션을 의미한다. 사용자가 필요로 하는 정보를 검색하여 보여주는 형태의 모바일 에이전트는 운영체제 위에서 다운로드 받은 파일을 내부의 애플리케이션과 인터페이스 관계를 형성하며 수행을 유도한다[1,4,8].

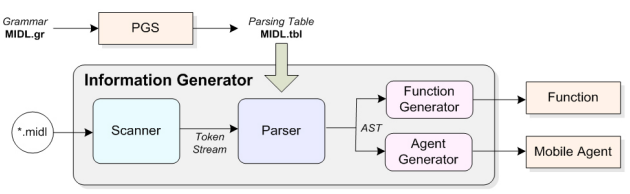
2.3 정보 생성기

정보 생성기란 MIDL 파일을 입력으로 받아 모바일 에이전트와 통합 에이전시에 등록할 함수를 생성하는 생성기를 말한다. 정보 생성기의 입력으로 사용되는 MIDL 파일은 CORBA 에서 사용하는 IDL 파일과 사용 측면에서 형식이 유사하지만 보다 간결한 문법으로 설계되었다. MIDL 파일에는 단 하나의 인터페이스가 존재하며 통합 에이전트에 등록될 함수들의 기본형이 기술된다. 함수의 기본형은 복귀 형과 함수이름, 매개변수들로 구성되며 인터페이스는 여러 개의 함수 기본형들을 기술할 수 있다. MIDL 의 EBNF 문법은 [표 1]과 같다.

[표 1] MIDL 의 EBNF 문법

```
<midl>          ::= <interface>
<interface>     ::= 'interface' <Identifier>
                    '{' <functionDef>
                    { <functionDef> } '}'
<functionDef>   ::= ('void'|'int'|'string')
                     <Identifier> '('
                     [ <formal_param> ] ')' ';'
<formal_param> ::= <param_dcl> { ','
                   <param_dcl> }
<param_dcl>     ::= 'int' | 'string'
```

이러한 MIDL 파일을 입력으로 받는 정보 생성기는 문법기법(Grammar Technique)을 이용하여 모바일 에이전트와 통합 에이전시에 등록할 함수를 생성한다. [그림 1]에서 보는 바와 같이 MIDL 파일을 입력으로 받아 어휘분석과 구문분석 과정을 거쳐 AST 를 생성한 후, 생성된 AST 를 운행하여 모바일 에이전트와 통합 에이전시에 등록할 함수를 생성한다[2,3,5].



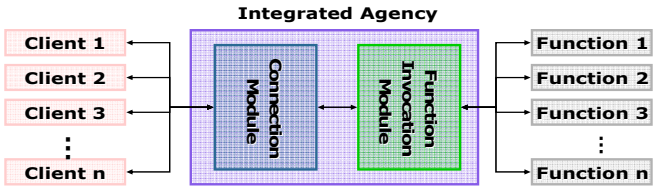
[그림 1] 정보 생성기의 구조

정보 생성기의 출력인 모바일 에이전트와 통합 에이전시에 등록할 함수는 사용자 정의 부분을 추가한 후, 모바일 장치를 위한 컨텐츠, 통합 에이전시의 함수 레지스트리, 그리고 함수 실행 환경에 등록된다.

3. 통합 에이전시

3.1 시스템 구성도

호스트 서버의 에이전트를 통합한 에이전시는 모바일 장치에 탑재된 가상 기계와 컨텐츠의 종류에 상관없이 단 하나의 에이전시를 통해 상호 운용이 가능하다. [그림 2]는 통합 에이전시의 전체 시스템 구성도를 도시한 것이다.

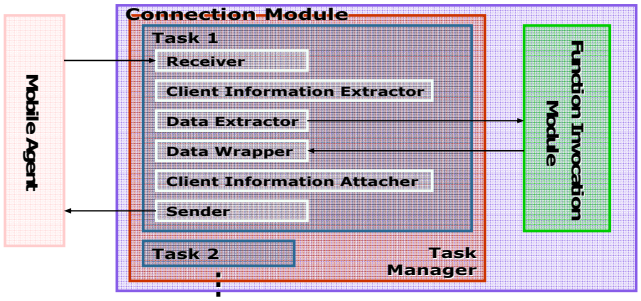


[그림 2] 시스템 구성도

통합 에이전시는 접속 모듈과 함수 호출 모듈로 구성되며 함수 생성 모듈에 해당하는 정보 생성기가 별도로 구성된다. 접속 모듈과 함수 호출 모듈에 대한 자세한 설명은 3.2 절과 3.3 절에서 설명하도록 하겠다.

3.2 접속 모듈

접속 모듈은 모바일 장치로부터 전송된 전송정보를 입력으로 받아 필요한 정보들을 추출하여 함수 호출 모듈로 전송하고 실행결과를 다시 모바일 장치로 전송하는 역할을 담당한다. 모바일 장치가 에이전시에 접속할 경우 접속 모듈의 작업 관리자는 새로운 작업을 생성한다. 각각의 작업들은 [그림 3]에 도시한 바와 같이 수신기, 클라이언트 정보 추출기, 데이터 추출기, 데이터 생성기, 클라이언트 정보 생성기, 송신기 컴포넌트로 구성된다.



[그림 3] 접속 모듈

3.2.1 수신기

수신기는 모바일 장치(모바일 에이전트)가 전송하는 전송정보를 수신하는 역할을 담당한다. 수신된 전송정보는 클라이언트 정보 추출기로 전송된다. XML 형태의 전송정보에 대한 스키마와 전송정보는 [표 2]에서 보는 바와 같이 <Client2Server>엘리먼트를 루트 엘리먼트로 가지며 <param>엘리먼트를 서브 엘리먼트로 가지는 <Content>엘리먼트가 존재한다. <Content>엘리먼트의 속성인 ID 는 모바일 장치의 가상 기계와 컨텐츠 ID 에 대한 정보를 제공한다. ID 속성은 모두 5 자리의 정수형 값으로 구성되며 가상 기계(1 자리), 컨텐츠 ID(4 자리)로 구성된다.

[표 2] Client2Server.xsd 와 전송정보

```
- Client2Server.xsd
<xsd:element name="Client2Server"
                        type="Client2ServerType">
 <xsd:complexType name="Client2ServerType">
 <xsd:element name="Content" type="ContentType">
  <xsd:complexType name="ContentType">
    <xsd:element name="param" minOccurs="0"
        maxOccurs="unbounded" type="xsd:string"/>
    <xsd:attribute name="name" type="xsd:string"/>
    </xsd:element>
  <xsd:attribute name="ID" type="xsd:string"/>
  </xsd:complexType>
 </xsd:element>
 </xsd:complexType>
</xsd:element>
- Client2Server.xml
<?xml version="1.0" encoding="euc-kr"?>
<Client2Server xmlns:xsi="http://www.w3.org/
                    2001/XMLSchema-instance"
                      xsi="client2server.xsd">
  <Content ID="20001">
    <param>1</param>
    <param>2</param>
  </Content>
</Client2Server>
```

3.2.2 클라이언트 정보 추출기

클라이언트 정보 추출기는 수신기로부터 전송된 전송정보 가운데 모바일 장치와 관련된 정보를 추출하는 역할을 담당한다. 모바일 장치의 정보는 모바일 장치에 탑재된 가상 기계의 명칭과 컨텐츠 ID 로서 모바일 장치에 설치된 모바일 에이전트에 의해 생성되는 정보이다. 추출된 모바일 장치에 대한 정보는 작업 매니저에 의해 임시 DB 에 저장된다. 모바일 장치에 대한 정보를 추출한 후, 나머지 전송 정보는 데이터 추출기로 전송된다.

3.2.3 데이터 추출기와 생성기

데이터 추출기는 모바일 장치로부터 전송받은 전송정보 가운데 모바일 장치에서 호출하고자 하는 함수의 매개변수에 대한 정보를 추출한다. 클라이언트 정보 추출기에서 추출된 컨텐츠 ID 와 데이터 추출기에서 추출된 매개변수는 함수 호출 모듈로 전송되어 해당 함수를 실행한다.

데이터 생성기는 함수 호출 모듈로부터 전송받은 실행결과를 미리 정의된 XML 형태의 전송정보 형태로 작성하는 컴포넌트이다.

3.2.4 클라이언트 정보 생성기

데이터 생성기로부터 전송받은 전송정보는 클라이언트에 대한 정보가 없는 상태이다. 따라서 미리 추출된 클라이언트에 대한 정보를 전송정보에 추가하여 클라이언트의 가상 기계와 컨텐츠에 맞는 전송정보 형태로 작성되어야 한다.

3.2.5 송신기

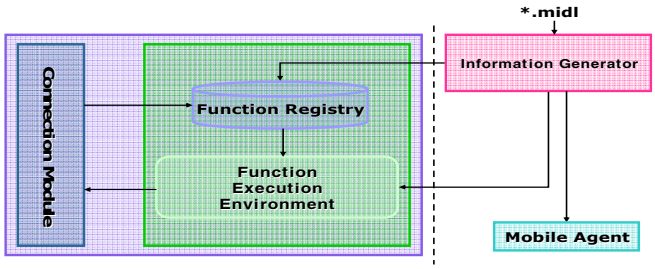
송신기는 호스트 서버로부터 모바일 장치로 전송정보를 전송하는 컴포넌트로서 모바일 에이전트로 전송할 전송정보가 모두 작성되면 송신기에 의해 모바일 에이전트로 전송정보가 전송되며, 전송정보의 스키마와 XML 형태의 전송정보는 [표 4]와 같다.

[표 4] Server2Client.xsd 와 전송정보

```
- Server2Client.xsd
<xsd:element name="Server2Client"
                        type="server2clientType">
 <xsd:complexType name="Server2ClientType">
  <xsd:element name="returnValue"type="returnValueType">
   <xsd:complexType name="returnValueType">
    <xsd:element name="value" type="xsd:string"/>
   </xsd:complexType>
  </xsd:element>
 </xsd:complexType>
</xsd:element>
- Server2Clinet.xml
<?xml version="1.0" encoding="euc-kr"?>
<Server2Client xmlns:xsi="http://www.w3.org/
                    2001/XMLSchema-instance"
                    xsi="client2server.xsd">
  <returnValue ID="20001">
    <value>10</value>
  </returnValue>
</Server2Client>
```

3.3 함수 호출 모듈

함수 호출 모듈은 접속 모듈로부터 전송되는 컨텐츠 ID 와 매개변수를 이용하여 실제 함수를 실행시키는 모듈로서 [그림 4]와 같이 함수 레지스트리와 함수 실행 환경으로 구성된다. 함수의 실행결과는 접속 모듈의 데이터 생성기 컴포넌트로 전송되어 모바일 장치로 전송된다.



[그림 4] 함수 호출 모듈

3.3.1 함수 레지스트리

함수 레지스트리는 정보 생성기로부터 생성되는 함수의 정보가 저장된다. 접속 모듈로부터 전송되는 컨텐츠 ID 가 함수 레지스트리에 존재할 경우, 함수 실행 환경을 통하여 해당 함수를 실행한다. 그러나 컨텐츠 ID 가 함수 레지스트리에 존재하지 않을 경우에는 오류 메시지가 접속 모듈로 전송된다. 이와 같은 함수 레지스트리의 형태는 [표 5]와 같다.

[표 5] 함수 레지스트리의 구조

ID	FunctionName	Params	ReturnType
0001	Othello	int;int	void
0002	Messenger	str	str

3.3.2 함수 실행 환경

함수 실행 환경은 일반적으로 알려진 멀티 스레드 시스템과 유사하다. 이와 같은 이유는 다수의 모바일 장치가 동시에 하나의 함수를 실행시키는 경우가 발생할 수 있기 때문이다. 따라서 함수의 실행은 공유 힙과 공유 데이터를 통하여 적은 부담을 갖는 문맥 전환을 통하여 멀티 스레드 방식으로 함수 실행이 이루어진다.

4. 결론 및 향후 연구 과제

모바일 장치와 통신기술의 발전, 그리고 모바일 장치를 위한 가상 기계의 등장은 셀룰러 폰으로 대표되는 모바일 장치의 기능을 향상시켰다. 모바일 장치의 기능이 향상됨에 따라 모바일 장치를 위한 다양한 컨텐츠들이 등장하였으며, 그 중에서도 특히 서버와의 통신을 통해 진행되는 Network 방식의 컨텐츠가 근래에 발표되는 컨텐츠 형식의 주류를 이루고 있다.

이와 같은 Network 방식의 컨텐츠는 에이전트가 설치된 별도의 서버를 필요로 한다. 그러나 모바일 장치에 탑재된 가상기계의 형식과 컨텐츠의 종류에 따라서 많은 수의 에이전트를 필요로 하는 단점을 가지고 있다.

본 논문에서는 이러한 문제점들을 해결하기 위한 방안으로서 여러 에이전트를 통합한 형태의 에이전시를 제시하였으며, 통합된 에이전시는 가상기계의 종류와 컨텐츠에 상관없이 단 하나의 에이전시를 통해 모바일 장치의 컨텐츠들과 통신이 가능하다. 또한 서로 다른 가상 기계 컨텐츠들 사이에서도 정보 교환이 가능하며, 유비쿼터스 컴퓨팅 환경을 위한 호스트 서버로의 역할도 수행할 수 있는 장점을 지닌다.

현재의 통합 에이전시는 일반적으로 알려진 모바일 장치를 위한 가상 기계만을 지원하고 있다. 따라서 모바일 장치를 위한 새로운 가상 기계가 등장할 경우, 수정이 불가피하다. 이러한 문제점을 해결하기 위한 방안으로서 새로운 가상 기계가 등장할 경우, 가상 기계의 특성을 고려하여 통합 에이전시를 수정할 수 있는 기법과 전체 통합 에이전시의 구성을 자동화할 수 있는 방안에 대한 향후 연구가 계속되어야 한다.

참고문헌

[1] 나승원, 오세만, "무선 환경에서 자원 관리를 위한 모바일 에이전트의 설계 및 구현", 정보과학회 춘계학술발표논문집, 제 30 권 제 1 호, pp.151~153, 2003.

[2] 오세만, 컴파일러 입문, 정익사, 2004.

[3] Alfred V.Aho, et al., Compilers Principle, Techniques, and Tools, Addison Wesley, 1988.

[4] Jim White, Mobile Agent White Paper, General Magic, 1996.

[5] Hongseob Shin, Seman Oh, "Design and Implementation of MIDL Compiler for Distributed Processing System in Mobile Environment," Proceedings of The 2005 International Symposium on Multimedia Applications in Edutainment, pp.43~48, 2005.

[6] Sunglim Yun, Eunbae Eom, Jungsook Kim, Seman Oh, "Design and Implementation of Wireless Payment System using GVM and MobileC," Proceedings of 2nd International Conference on Mobile Computing and Ubiquitous Computing, pp.36~45, 2005.

[7] T.Magedanz, "Intelligent Agent", CACM, Vol.37 No.7, pp.18~21, 1994.

[8] Xiaotong Zhuang, Santosh Pande, "Compiler Scheduling of Mobile Agents for Minimizing Overheads", Proceedings of The 23th ICDCS Conference, pp.600~609, 2003.