

자율적 반응과 목적성을 지닌 통신 개체

최근 인터넷의 성장과 더불어 주목 받고 있는 분야 중 하나는 에이전트 기술을 이용한 정보수집, 정보 관리, 정보 검색 등이다. 이 글에서는 에이전트의 기본 특성을 필두로 현재 출시된 에이전트와 연구 분야를 소개하고자 한다. 또한 실제 에이전트 개발시 사용되어지고 있는 프레임워크와 향후 웹과 에이전트의 발전 방향을 제시해 본다.

빈현식 / 넥스텔 연구원

에이전트 정의 및 특성

언제나 그렇듯이 어떤 대상에 대해서 설명할 때 제일 먼저 대상에 대한 정의를 내려야 하는데 실제로 에이전트에 대한 정의는 명확하지 않다. 따라서 지금까지 소개된 특성을 중심으로 설명하는 것이 이해를 돕는데 도움이 될 것이다. 에이전트에 대한 주요한 특성은 다음과 같다.

- **Autonomy:** 에이전트는 사람이나 다른 개체의 직접적인 간섭없이 작동하며, 내부적인 행동과 상태에 대한 제어기능을 갖는다.

- **social ability:** 에이전트는 다른 에이전트와 에이전트 통신언어를 통하여 통신한다.

- **reactivity:** 에이전트는 주어진 환경을 지각하고, 환경의 변화에 대해서 적절한 방법으로 반응한다.

- **Pro-activeness:** 에이전트는 단순히 환경에 반응하는 정도가 아니라 목적 지향적인 행동양식을 갖는다.

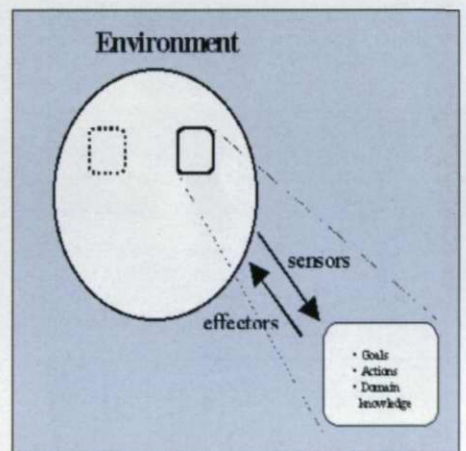
위와 같은 특성은 여러 책자나 저널에서 종종 나와서 더 이상 언급할 필요성이 없을 수도 있으나, 설명의 첫 장을 넘길 때 다시 한번 분명히 하는 것도 의미가 있을 것이다.

위에 언급한 특성 외에도 mobility, veracity, benevolence, rationality와 같은 특성을 나열할 수 있다. 필자는 위의 특성에 따라 에이전트에 대한 정의를 내리는 대신에 좀더 지명도 있는 전문가들의 견해를 빌리고자 한다.

- **Agent:** A person or thing that acts or is capable of action or is empowered to act, for another.

- **Software agent:** A computing entity that performs user delegated tasks autonomously.

그리고 필자의 견해를 더한다면 에이전트는 "주어진 환경에 대해서 자율적으로 반응하며, 목적성을 가지고 서로 통신하는 개체" 정도로 정의하고 싶다. 그러면,



(그림 1) 싱글에이전트 시나리오

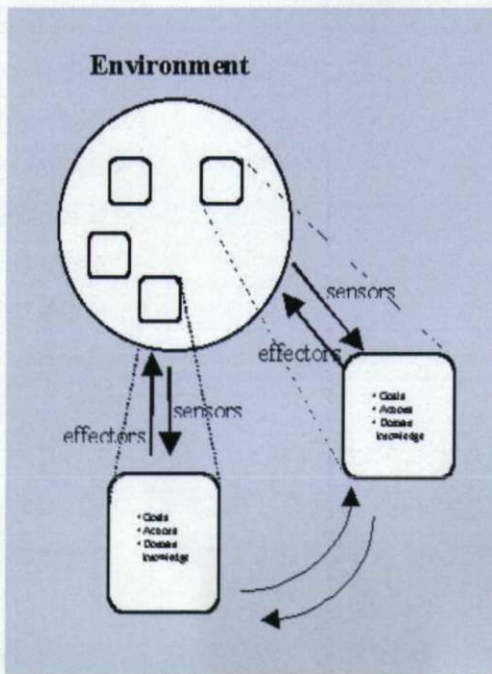
여기까지 에이전트에 대한 정의를 마무리 하겠다.

멀티 에이전트 시스템 소개

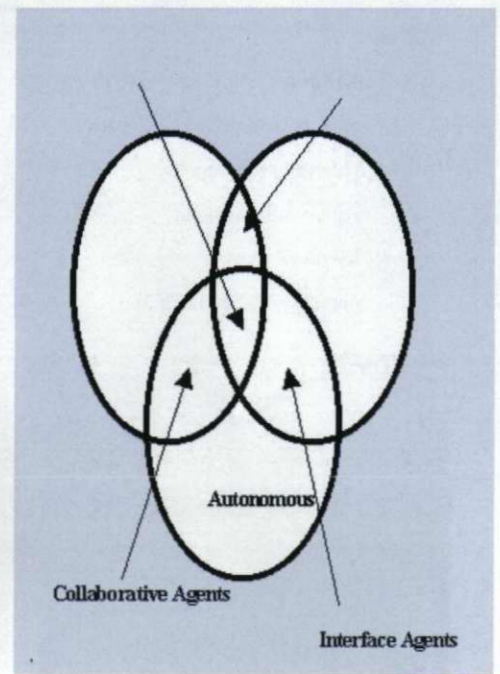
오늘날 멀티 에이전트 시스템 또는 에이전트 기반 패러다임은 분산화 된 컴퓨팅 시스템을 모델링하는데 유용하게 사용되어지고 있다. 그러나 대부분의 사람들이 집중화 된 싱글 에이전트에 익숙해 있으므로, 멀티에이전트와의 구분을 잠시 언급하겠다.

싱글에이전트 시스템에서의 에이전트는 어떠한 다른 개체도 에이전트로 인식하지 않으며, 따라서 실제로 다른 에이전트가 존재한다고 할지라도 단지 환경(environment)의 일부로 간주한다.

반면, 멀티에이전트 시스템은 각각의 에이전트가 서로간의 목적과 행동을 모델링하며, 환경의 변화가 다른 에이전트에 의해서 결정될 수 있다는 측면에서 싱글에이전트 시스템과 차별화된다.



〈그림 2〉 멀티에이전트 시나리오



〈그림 3〉 에이전트 Typology

멀티 에이전트의 이해를 돕기 위해 리서치 그룹 모델링의 예를 제시한다. 리서치 그룹은 가설을 확인하거나 증명하기 위해 형성된 그룹을 의미하며, 가설이 확인될 경우에는 consensual 지식베이스에 합치될 수 있다. 토론에 참석한 모든 리서처들은 지식베이스에 의해서 특정한 물음 가지며 아래와 같은 7개의 룰로 구성되어 질 수 있을 것이다.

- Researcher: 연구활동 수행, 아이디어 도출, 토론 참가, 다른 리서치그룹과의 접촉 등의 활동을 하는 그룹의 구성원이다.
- Assistants: 실질적으로 토론에 참가하지는 않으나 리서처의 요구에 따라 결과를 리서처에게 보고하는 그룹의 구성원이다.
- Facilitator: 리서처 그룹 멤버의 정보를 가지고 멤버간의 접촉을 담당하며, 그룹의 멤버가 아니다.
- Author: 확인을 위해서 가설을 제안

한 리서처이다. 즉, author는 consensual 지식베이스를 풍부하게 할 수 있는 지식을 가지고 있다.

- Mediator: author와 critics의 수를 협상하고, author와 critics간의 협상을 중재한다
- Critic: 제안된 가설을 판단하고 평가하며 검증하기 위해 author에 의해 선택된 그룹의 리서처이다.
- Bookkeeper: 그룹의 consensual 지식베이스에 저장된 모든 정보를 관리하는 리서처이다.

에이전트 응용 소프트웨어 소개

에이전트의 이해를 돕기 위해서는 어떤 분야에서 어떻게 사용되고 있는지를 살펴볼 필요성이 있다.

- 지능형 에이전트를 이용한 재해대책
지바공업대학은 에이전트라 불리는 소프트웨어의 사고 방식을 도입해 지진 등

의 재해시 인간의 행동을 정확하게 예측하는 소프트웨어를 개발했다. 에이전트는 각자가 독립해 자율판단 기능을 갖는 소프트웨어라고 할 수 있다. 피난자 한 사람 한 사람의 재해시의 판단을 에이전트에게 담당시켜 집단으로의 피난 행동을 분석하고 있다. 피해 예측을 한다든지 피난 장소를 시가지에 효과적으로 설치할 때에도 도움이 될 것이다.

- 에이전트를 이용한 정보검색
NTT 데이터 통신은 정보가 발견될 가능성이 높은 홈페이지를 판단해 단 시간 내에 결과를 알려주는 에이전트의 실용화 시험을 했다. 인터넷을 경유하는 데이터베이스 검색 시스템을 구축해 '인터페이스', '검색프란너', '데이터베이스'의 3가지의 에이전트를 설정했다.

인터페이스 에이전트는 사용자의 입력 키워드를 받아 구분하여 SGML이라 불리는 서식으로 표기하여 둔다. 프란너는 이 표시를 근거로 워크스테이션 내에 미리

```
(ask-one
  : sender joe
  : content (real price = sun.price())
  : receiver stock-server
  : reply-with sun-stock
  : language java
  : ontology NYSE-TICKS)
```

(그림 4) KQML예제

기억하고 있는 약 7천의 항목과 대응시킨다. 프란너는 항목마다 관련이 깊은 데이터베이스를 기억하고 있어 대응하는 데이터베이스 에이전트에 연락한다. 연락을 받은 에이전트는 검색 요구를 결정된 방식으로 개선해 요구되는 정보를 끌어낸다.

• 코치

코치시스템은 사용자가 리스프 프로그램을 작성하는 작업형태를 관찰해 적응

사용자 모델을 만들고 작업지식과 적응사용자 모델에 따라 사용자가 앞으로 하려는 일을 추론해 능력에 맞는 도움말을 제시해 주는 역할을 한다. 코치 시스템은 학습기능과 추론기능을 갖는 지능형 에이전트이며, 컴퓨터 작업환경을 돕는다는 측면에서는 사용자 인터페이스 에이전트이다. 이와 같이 지능형 에이전트는 사용자 인터페이스 에이전트와 밀접한 관계가 있다.

텔레스크립트(Telescript)

제너럴 매직의 텔레스크립트는 에이전트 전문 언어로 이동 에이전트가 갖춰야 할 대부분의 개념을 정립, 구현한 것이다. 에이전트 생성이나 이동이 모두 텔레스크립트로 기술되며, 그 언어는 텔레스크립트 엔진에 의해 해석되고 수행된다.

텔레스크립트 엔진은 개인적인 통신을 위해 사용하는 것과 공공 정보 서비스를 위해 사용되는 것, 두 분류로 나뉘 제공하고 있다. 초기에 텔레스크립트는 PDA와 연관돼 많은 연구가 진행됐지만, 현재는 웹을 충분히 활용할 수 있는 툴로 확장되어 있는 상태이다.

그 외에도 제너럴 매직은 웹과 관련해 타브리즈 에이전트웨어와 타브리즈 에이전트틀로 나누어져 있다. 타브리즈 에이전트웨어는 에이전트를 생성해 상호 작용하며 에이전트 프로세스를 이용해 정보 교환, 인증, 검증하는 절차를 수행한다. 개발 환경인 타브리즈 에이전트 툴은 소스 코드 에디터, 디버거, 클래스 브라우저, 에이전트 매니저 등을 갖고 있다.

KQML

KQML은 자율적인 정보 시스템간의 지식을 공유하기 위한 프로토콜의 개발노력에 대한 중요한 결과이다. KQML은 1990년에 설립된 DARPA Knowledge External Interfaces Working Group에 의한 것이며, 이외에도 Knowledge Interchange Format (KIF) working group, Ontologies working group, Knowledge Representations Systems Standards (KRSS) working group 등이 있다. KQML은 에이전트들간에 지식 또는 정보를 공유할 수 있도록 하는 프레임워크를 제공한다. KQML 메시지는

```
<performative> ::= (<word> {<whitespace> :<word> <whitespace> <expression>}*)
<expression> ::= <word> | <quotation> | <string> |
                 (<word> {<whitespace> <expression>}*)
<word> ::= <character><character>*
<character> ::= <alphabetic> | <numeric> | <special>
<special> ::= < | > | = | + | - | * | / | # | ^ | ~ | ` | _ |
              @ | $ | % | | : | . | | | | 7
<quotation> ::= '<expression>' | '<comma-expression>'
<comma-expression> ::= <word> | <quotation> | <string> | ,<comma-expression>
                    (<word> {<whitespace> <comma-expression>}*)
<string> ::= "<stringchar>*" | #<digit><digit>*"<ascii>"
<stringchar> ::= \<ascii> | <ascii>-\-<double-quote>
```

(그림 5) KQML String Syntax in BNF

(표 1) Summary of reserved parameter keywords and their meanings

Keyword	Meaning
:content	the information about which the performative expresses an attitude
:force	whether the sender will ever deny the meaning of the performative
:in-reply-to	the expected label in a reply
:language	the name of representation language of the :content parameter
:ontology	the name of the ontology (e.g., set of term definitions) used in the :content parameter
:receiver	the actual receiver of the performative
:reply-with	whether the sender expects a reply, and if so, a label for the reply
:sender	the actual sender of the performative

performatives라고도 하며, 이것을 사용하여 다른 에이전트에게 정보를 질의하거나, 정보를 말하거나하는 일련의 행동들을 할 수 있다. KQML은 정보를content, message, communication 등의 3단계의 다른 레벨로 인코딩한다. 제시된 예에서 "ask-one"은 performative의 이름을 표시하고, "content"라고 쓰여진 것은 content레벨쪽에 해당하고, "language", "ontology"는 message레벨쪽, "sender", "receiver", "reply-with"는 communication레벨에 해당한다.

여기에 KQML 스티링 구문법과, 예약된 performative 변수들, 예약된 performatives을 나열한다.

자바 기반 에이전트 환경

자바가 지능형 에이전트를 구현하기에 가장 적합한 언어는 아니라고 할지라도 적용 가능한 언어임은 틀림이 없을 것이다. 자율적인 측면에서 자바 어플리케이션은 지속적으로 실행될 수 있는 독립적인 프로세스이거나 쓰레드일 수 있으며, 다른 프로그램과 소켓 등의 방법으로 통신을 지원한다.

지능적인 측면에서 또한 초기의 AI 프로그램들이 LISP나 Prolog로 쓰여졌으나, 최근에는 C 또는 C++로 쓰여졌으며, 따라서 자바로도 쓰여질 수 있음의 미한다. 지식의 표현은 자바클래스의 속성으로 표현되어질 수 있을 뿐 아니라, 다른 개체에 대한 링크 또는 레퍼런스를 가질 수 있다.

이동성이라는 측면에서 자바의 이동 가능한 바이트 코드 및, JAR 파일은 네트워크를 통하여 다른 목적 머신에서 실행될 수 있으며, 자바 애플릿은 웹 브라우저를

(표 2) Summary of reserved parameter keywords and their meanings

Name	Section	Meaning
achieve	5.6	S wants R to do make something true of their environment
advertise	5.8	S is particularly-suited to processing a performative
ask-about	5.4	S wants all relevant sentences in R's VKB
ask-all	5.4	S wants all of R's answers to a question
ask-if	5.4	S wants to know if the sentence is in R's VKB
ask-one	5.4	S wants one of R's answers to a question
break	5.10	S wants R to break an established pipe
broadcast	5.10	S wants R to send a performative over all connections
broker-all	5.11	S wants R to collect all responses to a performative
broker-one	5.11	S wants R to get help in responding to a performative
deny	5.1	the embedded performative does not apply to S (anymore)
delete	5.2	S wants R to remove a ground sentence from its VKB
delete-all	5.2	S wants R to remove all matching sentences from its VKB
delete-one	5.2	S wants R to remove one matching sentence from its VKB
discard	5.7	S will not want R's remaining responses to a previous performative
ecs	5.5	end of a stream of responses to an earlier query
error	5.3	S considers R's earlier message to be mal-formed
evaluate	5.4	S wants R to simplify the sentence
forward	5.10	S wants R to route a performative
generator	5.7	same as a standby of a stream-all
insert	5.2	S asks R to add content to its VKB
monitor	5.9	S wants updates to R's response to a stream-all
next	5.7	S wants R's next response to a previously-mentioned performative
pipe	5.10	S wants R to route all further performatives to a another agent
ready	5.7	S is ready to respond to R's previously-mentioned performative
recommend-all	5.11	S wants all names of agents who can respond to a performative
recommend-one	5.11	S wants the name of an agent who can respond to a performative
recruit-all	5.11	S wants R to get all suitable agents to respond to a performative
recruit-one	5.11	S wants R to get another agent to respond to a performative
register	5.10	S can deliver performatives to some named agent
reply	5.4	communicates an expected reply
rest	5.7	S wants R's remaining responses to a previously-mentioned performative
sorry	5.3	S cannot provide a more informative reply
standby	5.7	S wants R to be ready to respond to a performative
stream-about	5.5	multiple-response version of ask-about
stream-all	5.5	multiple-response version of ask-all
subscribe	5.9	S wants updates to R's response to a performative
tell	5.1	the sentence in S's VKB
transport-address	5.10	S associates symbolic name with transport address
unregister	5.10	a deny of a register
untell	5.1	the sentence is not in S's VKB

통하여 리모트에서 또한 실행될 수 있다. 따라서 인터넷의 발전과 더불어 자바와 에이전트의 관계를 고찰하는 것 또한 대단히 의미있는 주제가 될 것이다. 따라서 필자는 아래에서 자바에 기반한 에이전트 기반 환경을 고찰한다.

• JATLite

JATLite는 스탠포드 대학에서 멀티에이전트시스템으로 사용되어질 수 있도록 만든 프레임워크로서 순수 자바로 쓰여져 있다. 구조는 5개의 레이어로 구성되어 있으며 각 레이어에 대한 설명은 다음과 같다.

- Abstract 레이어

상위의 클래스들이 정의될 수 있도록 하는 루트 클래스들을 제공한다.

- Base 레이어

TCP/IP를 이용한 통신 레이어이며, 에이전트가 메시지 프로토콜을 정의하도록 지원한다

- KQML 레이어

JATLite는 KQML의 완전한 구문을 제공하며, 에이전트가 KQML 통신을 하도록 지원한다.

- Router 레이어

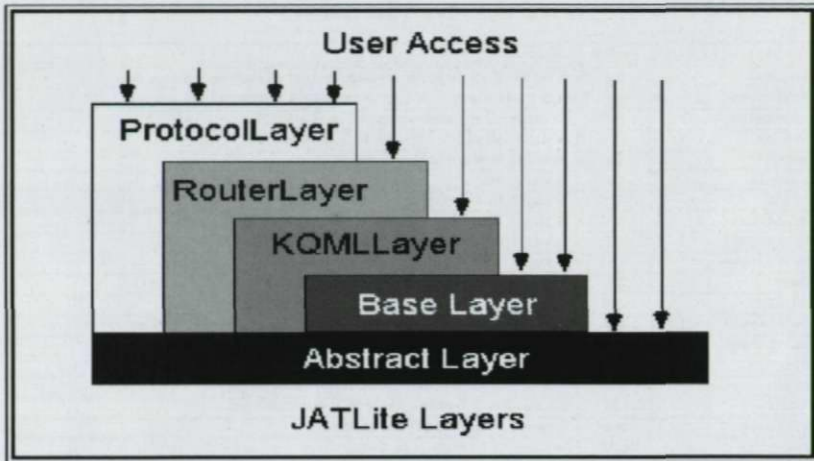
메시지 라우팅을 지원한다.

- Protocol 레이어

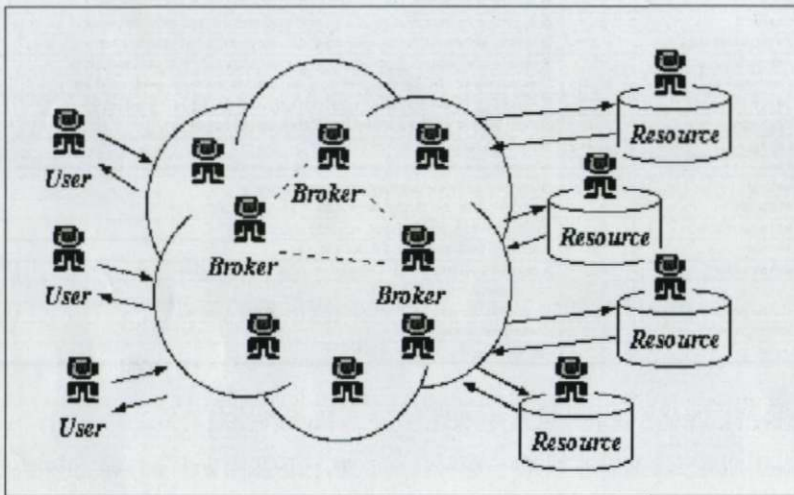
SMTP와 FTP와 같은 표준 인터넷 서비스를 지원한다.

• Aglets

aglet은 IBM에서 개발한 자율적인 자바기반 에이전트로 이동성에 대한 능력을



(그림 6) Java Agent Template Lite Layers



(그림 7) Infosleuth 시나리오

제공하며 이동해야 할 목적지와 각 위치에서 수행해야 할 행동을 정의할 수 있다. aglet 호스트 어플리케이션이 설치되어진 시스템에서 플랫폼에 독립적으로 실행될 수 있으며, 보안 관리자를 포함한다. 자세한 내용은 <http://www.trl.ibm.co.jp/aglets/>을 참조하면 된다.

• Voyager

ObjectSpace의 Voyager는 순수자바로 쓰여졌으며, 에이전트를 가능하게 하는 확장된 Object Request Broker (ORB)로 베이스 클래스에서 제공되는 이동성과 자

울성을 갖는다. aglet과 마찬가지로 특정 위치에서 수행할 행동을 정의할 수 있도록 하며, 보안 관리자를 가지고 있다. 자세한 내용은 <http://www.objectspace.com/>을 참고하면 된다.

• Odyssey

General Magic의 Odyssey는 순수자바로 쓰여졌으며, JDK 1.1을 필요로 하며, 트랜스포트로서 RMI를 사용하였으나, DCOM과 IIOP를 사용할 수 있도록 구성되어질 수 있다. 중요한 클래스로서 Agent, Worker, Ticket를 가지고 있으며,

Agent나 Worker를 상속 받음으로써 에이전트가 될 수 있다.

• InfoSleuth

InfoSleuth는 이질적인 데이터 소스로부터 정보를 찾고, 모으고, 재구성하기 위해서 에이전트를 사용하는 MCC의 연구 프로젝트이다. KQML, KIF, HTTP와 자바에 기반하여 구성되어 있으며, 멀티 에이전트의 기반에서 정보를 모으고 관리한다.

• ABE

IBM의 ABE는 어플리케이션에 지능형 에이전트를 채용하기 위한 C++ 라이브러리 또는 아키텍처라고 할 수 있으며, ABE의 부분으로 구성된 순수 자바인 CIAgent가 있다. ABE는 자바 어댑터를 통해서 자바 어플리케이션과 통신할 수 있다. 다른 에이전트 환경과 달리 '인텔리전트'에 중점을 두었으며, Engines, Knowledge, Library, Views, Adapters로 구성되어져 있다.

정보 형태 분류와 웹 문서의 변화

여기까지 에이전트의 기본적인 내용과 구현시 유용하게 사용되어질 수 있는 몇 가지 프레임워크를 살펴보았다. 다소 추상적이고, 딱딱하고 어려운 면이 없지 않았을 것이다. 그럼, 이제부터 웹에 존재하는 여러 가지 정보를 에이전트를 이용하여, 기존과는 다른 방법으로 찾는 방법을 소개하겠다.

기존의 정보를 잘 정제하는 방법은 여러 가지가 있겠으나, 현재까지 데이터의 관리는 고전적으로 IR(Information Retrieval)과 DBMS로 발전해왔다고 할

수 있다. 이것에 대한 설명을 더 진행하기 전에 문서의 종류를 형태상으로 분류해보면 아래와 같다.

• structured 정보

잘 구조화된 정보를 얘기한다. 예를 들어 테이블형태로 잘 정의되어서 스키마를 정의한 후 데이터베이스에서 관리되어질 수 있는 문서들이다.

• semi-structured 정보

완벽하게 스키마를 정의할 수는 없으나, 문서에 포함되어져 있는 태그 정보를 이용하여, 부분적인 데이터 관리가 가능한 문서이다. 이러한 문서에는 HTML, LATEX등이 있을 수 있다.

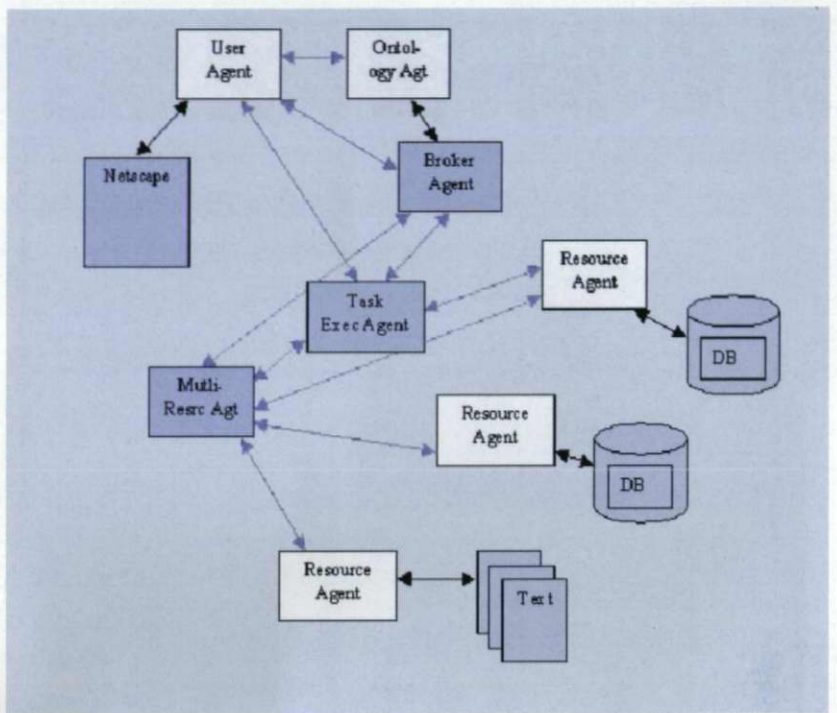
• unstructured 정보

구조를 파악할 수 없는 문서들이며, 이러한 문서에는 텍스트 파일, 워드 파일 등이 있을 수 있다.

이때 까지 고전적인 방법으로는 structured 정보는 DBMS를 통하여 관리되어졌으며, semi-structured 정보와 unstructured 정보에 대해서 IR에 의해서 검색되어졌다.

현재는 XML이라는 것이 나와서 많은 관심을 모으고 있다. XML을 데이터적인 측면에서 HTML과 구분하는 가장 중요한 것은 그 자체에 의미 태그를 가지고 있다는 것이다.

일반적으로 태그는 포맷정보와 의미정보를 가지는 것으로 구분되며, HTML이 포맷정보를 가지는 것에 치중했으므로, 문서의 관리가 일반적인 IR에 한정되어있었으나, XML은 스키마가 정의된 DBMS에 의해서 관리되어질 수 있는 가능성을



(그림 8) 간략한 Infosleuth Network

제공함으로써 혁신적인 데이터 관리의 발전이 예상된다.


정보분석 에이전트

1980년도 한국에 인터넷이 보급되기 시작한 이래, 웹브라우저의 등장으로 일반인들에게 웹문서에 쉽게 접근할 수 있는 방법을 제공하였다. 그러나 곳곳에서 산발적, 기하급수적, 비구조적으로 만들어진 문서는 날이 갈수록 방대해지고 있다.

따라서 이처럼 산재해 있는 많은 정보들 가운데, 사용자가 꼭 필요로 하는 정보를 신속하고, 정확하게 찾아내는 기술들이 점점 중요시 되고 있다. 현재 단순한 정보검색의 단계를 넘어 직접 정보를 찾지 않아도 원하는 정보를 제공해주는 뉴스배달시스템이나 푸시 등 많은 시스템이 출현된 상태이다.

위에서 잠시 언급한 바가 있는 Infos-

leuth시스템을 다시 살펴보기로 한다. Infosleuth시스템은 MCC에서 진행하는 프로젝트에 의한 것이며, 멀티에이전트 기반하에서 정보수집, 정보 분류, 정보 검색 등을 지원하며, KQML을 통해 에이전트간 통신을 수행한다.

Semi-structured 정보를 수집하여, 통계적인 데이터 마이닝과 추론을 통하여 데이터를 분석하고, 사용자에게 따른 다른 검색이 가능하도록 되어있다. 도메인에 따른 ontology관리기와 사용자에게 그 그래픽 유저 인터페이스를 제공한다. 

데이터베이스월드

URL:www.dpc.or.kr