



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년01월17일
(11) 등록번호 10-1008977
(24) 등록일자 2011년01월11일

(51) Int. Cl.

G06F 11/00 (2006.01)

(21) 출원번호 10-2004-0012746
(22) 출원일자 2004년02월25일
심사청구일자 2009년02월06일
(65) 공개번호 10-2005-0087129
(43) 공개일자 2005년08월31일
(56) 선행기술조사문헌
KR100175455 B1
KR100334904 B1

(73) 특허권자

삼성전자주식회사

경기도 수원시 영통구 매탄동 416

(72) 발명자

정성원

서울특별시성동구옥수2동극동아파트8동1211호

장영숙

경기도의정부시호원동신도6차아파트602-1402호

(뒷면에 계속)

(74) 대리인

정상빈, 특허법인가산

전체 청구항 수 : 총 9 항

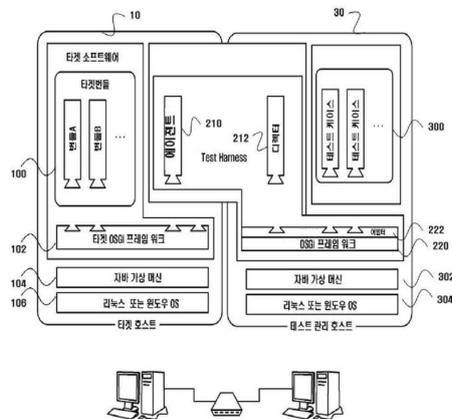
심사관 : 이경홍

(54) OSGi 서비스 플랫폼 테스트 방법 및 이를 이용한 테스트 툴

(57) 요약

본 발명은 OSGi 서비스 플랫폼 테스트 방법 및 이를 이용한 테스트 툴에 관한 것으로서, 더욱 상세하게는 OSGi 서비스 플랫폼의 테스트 툴에 포함되는 번들의 구성과 기능을 변경하여 효율적으로 서비스 번들의 동작여부를 테스트 할 수 있는 방법 및 이를 이용한 테스트 툴에 관한 것이다. 본 발명의 실시예에 따른 OSGi 서비스 플랫폼을 테스트 툴은 테스트 케이스들이 설치되는 프레임워크; 상기 프레임워크를 테스트 실행 환경의 프레임워크로 동작하게 하는 어댑터; 테스트 툴의 UI와 테스트를 관리할 수 있는 서비스 및 OSGi 서비스들을 제공하는 디렉터 번들; 및 타겟 호스트의 프레임워크에 테스트 케이스 번들과 메시지의 이동을 중계하는 에이전트 번들을 포함한다.

대표도 - 도3



(72) 발명자

여기대

서울특별시동작구흑석1동165-7

성숙희

서울특별시관악구봉천본동두산아파트102-904호

이현동

경기도수원시팔달구영통동1047-1건영아파트423

동1801호

특허청구의 범위

청구항 1

테스트 관리 호스트의 프레임워크에,

테스트 툴의 UI와 테스트를 관리할 수 있는 서비스 및 OSGi 서비스들을 제공하는 디렉터 번들을 설치하고,

타겟 호스트의 프레임워크에,

테스트 케이스 번들과 메시지의 이동을 중계하는 에이전트 번들과 테스트할 서비스 번들을 설치하는 단계;

상기 에이전트 번들이 상기 디렉터 번들로부터 전달된 소정의 테스트 케이스를 상기 타겟 호스트의 프레임워크에 설치하여 테스트를 수행하는 단계; 및

상기 테스트 결과 및 테스트 로그를 테스터에게 UI 또는 파일로 제공하는 단계를 포함하는 OSGi 서비스 플랫폼 테스트 방법.

청구항 2

제 1항에 있어서, 상기 설치된 테스트 케이스 번들과 서비스 번들은 UI 또는 스크립트를 통해 입력받을 수 있는 OSGi 서비스 플랫폼 테스트 방법.

청구항 3

제 1항에 있어서, 상기 테스트 로그가 텍스트 형식의 파일로 제공되는 OSGi 서비스 플랫폼 테스트 방법.

청구항 4

테스트 케이스 번들이 설치되는 프레임워크;

상기 프레임워크를 테스트 실행 환경의 프레임워크로 동작하게 하는 어댑터; 및

테스트 툴의 UI와 상기 테스트를 관리할 수 있는 서비스 및 OSGi 서비스들을 제공하고, 타겟 호스트의 프레임워크에 상기 테스트 케이스 번들과 메시지의 이동을 중계하기 위하여 상기 타겟 호스트에 구비된 에이전트 번들과 통신하는 디렉터 번들을 포함하는 OSGi 서비스 플랫폼 테스트 장치.

청구항 5

제 4항에 있어서, 상기 디렉터 번들은

모듈간 메시지 전달을 담당하는 핸들러 모듈;

테스트 케이스 번들 및 테스트 로그들을 관리하는 테스트 관리 모듈;

상기 에이전트 번들과 내부 모듈의 패킷 전달을 담당하는 네트워크 패킷처리 모듈;

상기 에이전트 번들의 타겟 호스트의 정보를 감지하는 타겟 디스커버리 모듈;

상기 타겟 호스트 정보를 처리하는 타겟관리 모듈;

번들의 상태 변경 이나 설치 작업을 하는 OSGi 번들 라이프사이클 관리 모듈;

GUI를 통한 테스터의 요구를 처리하여 핸들러 모듈에게 전달하는 GUI 처리 모듈; 및

콘솔 창에서 입력되는 테스터의 커맨드를 처리하는 콘솔 커맨드 처리 모듈을 포함하는 OSGi 서비스 플랫폼 테스트 장치.

청구항 6

제 5항에 있어서, 상기 에이전트 번들은

타겟 호스트의 정보에 대한 통보 메시지를 상기 디렉터로 전송하고, 디렉터를 관리하는 디렉터 관리 모듈;

테스트 케이스 번들을 프레임워크에 설치하고 테스트 케이스 번들로부터 받은 로그와 테스트 결과를 디렉터에게

전달하는 기능을 수행하는 테스트 케이스 관리 모듈;

네트워크로 받은 패킷을 처리해서 에이전트로 전송하는 네트워크 패킷 처리 모듈; 및

테스트 로그를 처리하여 출력하는 콘솔 커맨드 처리 모듈을 포함하는 OSGi 서비스 플랫폼 테스트 장치.

청구항 7

제 6항에 있어서, 상기 설치된 테스트 케이스 번들과 서비스 번들은 UI 또는 스크립트를 통해 입력받을 수 있는 OSGi 서비스 플랫폼 테스트 장치.

청구항 8

제 7항에 있어서, 상기 디렉터 번들의 콘솔 커맨드 처리 모듈은 테스트 진행 중에 발생한 오류에 대응할 수 있는 커맨드 명령을 입력할 수 있는 수단을 제공하는 OSGi 서비스 플랫폼 테스트 장치.

청구항 9

제 8항에 있어서, 상기 테스트 로그가 텍스트 형식의 파일로 제공되는 OSGi 서비스 플랫폼 테스트 장치.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- [0027] 본 발명은 OSGi 서비스 플랫폼 테스트 방법 및 이를 이용한 테스트 툴에 관한 것으로서, 더욱 상세하게는 OSGi 서비스 플랫폼의 테스트 툴에 포함되는 번들의 구성과 기능을 변경하여 효율적으로 서비스 번들의 동작여부를 테스트 할 수 있는 방법 및 이를 이용한 테스트 툴에 관한 것이다.
- [0028] 도1은 종래의 OSGi.org에서 제공하고 있는 OSGi 서비스 플랫폼 테스트 툴의 전체 구조를 보여주는 도면이다.
- [0029] 종래의 OSGi.org에서 제공하고 있는 OSGi 서비스 플랫폼 테스트 툴은 도 1에서 도시한 바와 같이, 타겟 호스트(10, 테스트 받을 호스트)에 위치하는 osgi-refimpl (reference implement) 번들(200) 및 타겟번들(202)과 테스트 관리 호스트(30)(Test Manager Host)에 위치하는 디렉터(Director)번들(204) 및 서포트(Support)번들(206)로 이루어진다. 이때, osgi-refimpl(reference implement) 번들(200), 타겟 번들(202), 디렉터 번들(204) 및 서포트 번들(206)은 각각 도2a, 도2b, 도2c, 도2d에서 도시한바와 같이 복수개의 번들들로 구성된다.
- [0030] 그런데 OSGi.org에서 제공하는 테스트 툴은 타겟호스트 쪽에 refimpl 번들이 타겟 번들과 따로 설치되어 구조가 복잡하고, OSGi 서비스 번들을 추가하려면 진행중인 테스트를 중단하고 refimpl 번들의 압축을 풀어서 새로 개발한 OSGi 서비스 번들로 교체하고 다시 JAR 파일로 만든후 테스트를 진행시켜야 하는 문제가 있었다.
- [0031] 또한, 테스트 관리 호스트(30)에 서포트 번들이 디렉터 번들과 따로 설치되어 구조가 복잡하고, OSGi.org 프레임워크부터 테스트 케이스의 설치까지 target.bat 파일에 의해 자동으로 진행되어 테스트 케이스 번들을 추가하려면 테스트 케이스 번들 전체를 묶은 테스트 번들의 압축을 풀어서 테스트 번들의 번들 정보를 수정하고 다시 JAR 파일로 만들어야 하는 과정이 필요하기 때문에 임의의 시점에서 추가 테스트 케이스를 추가할 수 없고, 또한 테스트 케이스 번들을 통합하지 않고 독립적으로 실행시킬 수 없는 문제가 있었다.
- [0032] 다음으로 테스트 툴 GUI에 문제가 발생하였을 때 문제의 원인을 분석하기 위해 진행중인 테스트 과정을 강제 종료시켜야 했고, 테스트 로그가 바이트(byte) 스트리밍(streaming) 포맷 형태의 파일로 저장되기 때문에 테스트가 진행중이 아니면 저장된 로그를 일반 에디터(editor) 프로그램에서 보기가 어려운 문제가 있었다.

발명이 이루고자 하는 기술적 과제

- [0033] 본 발명은 OSGi 테스트 툴의 번들을 통합하여 구조의 복잡성을 개선하고, 테스트 케이스나 테스트할 OSGi 서비스 번들의 변경을 편리하게 할 수 있는 방법을 제공하는데 그 목적이 있다.
- [0034] 본 발명의 또 다른 목적은 테스트 과정에서 오류가 발생하는 경우 오류 원인의 분석과 테스트 후 테스트 로그

(Log)의 분석을 용이하게 할 수 있는 방법을 제공하는 것이다.

[0035] 본 발명의 목적들은 이상에서 언급한 목적들로 제한되지 않으며, 언급되지 않은 또 다른 목적들은 아래의 기재로부터 당업자에게 명확하게 이해되어질 수 있을 것이다.

발명의 구성 및 작용

[0036] 상기 목적을 달성하기 위하여, 본 발명의 실시예에 따른 OSGi 서비스 플랫폼을 테스트 툴은 테스트 케이스들이 설치되는 프레임워크; 상기 프레임워크를 테스트 실행 환경의 프레임워크로 동작하게 하는 어댑터; 테스트 툴의 UI와 테스트를 관리할 수 있는 서비스 및 OSGi 서비스들을 제공하는 디렉터 번들; 및 타겟 호스트의 프레임워크에 테스트 케이스 번들과 메시지의 이동을 중계하는 에이전트 번들을 포함한다.

[0037] 본 발명의 실시예에 따른 OSGi 서비스 플랫폼을 테스트 방법은 테스트 관리 호스트의 프레임워크에, 테스트 툴의 UI와 테스트를 관리할 수 있는 서비스 및 OSGi 서비스들을 제공하는 디렉터 번들을 설치하고, 타겟 호스트의 프레임워크에, 테스트 케이스 번들과 메시지의 이동을 중계하는 에이전트 번들과 테스트할 서비스 번들을 설치하는 단계; 상기 에이전트 번들이 상기 디렉터 번들로부터 전달된 소정의 테스트 케이스를 상기 타겟 호스트의 프레임워크에 설치하여 테스트를 수행하는 단계; 및 상기 테스트 결과 및 테스트 로그를 테스터에게 UI 또는 파일로 제공하는 단계를 포함한다.

[0038] 기타 실시예들의 구체적인 사항들은 상세한 설명 및 도면들에 포함되어 있다.

[0039] 본 발명의 이점 및 특징, 그리고 그것들을 달성하는 방법은 첨부되는 도면과 함께 상세하게 후술되어 있는 실시예들을 참조하면 명확해질 것이다. 그러나 본 발명은 이하에서 개시되는 실시예들에 한정되는 것이 아니라 서로 다른 다양한 형태로 구현될 수 있으며, 단지 본 실시예들은 본 발명의 개시가 완전하도록 하고, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 발명의 범주를 완전하게 알려주기 위해 제공되는 것이며, 본 발명은 청구항의 범주에 의해 정의될 뿐이다. 명세서 전체에 걸쳐 동일 참조 부호는 동일 구성 요소를 지칭한다

[0040] 이하, 첨부된 도면을 참조하여 본 발명의 바람직한 실시예를 상세히 설명하기로 한다.

[0041] 도 3은 본 발명의 일실시예에 따른 OSGi 서비스 플랫폼 테스트 툴의 전체 구조를 보여주는 도면이다.

[0042] 보다 상세하게 본 발명의 일실시예에 따른 OSGi 서비스 플랫폼 테스트 툴의 테스트 관리 호스트(30)에서의 구성은 도 4에서 도시한 바와 같이, OSGi 프레임워크(220)와 어댑터(222)와 시스템 번들(213)과 디렉터 번들(212) 및 상기 디렉터 번들(212)에 의해 로드되어 지는 서비스들로 구성된다. 이때, OSGi 프레임워크(220)에 동적으로 로드되는 다른 번들들과 OSGi 프레임워크(220)의 환경을 제공하는 자바가상머신(303, Java Virtual Machine)은 외부 컴포넌트가 된다.

[0043] 상기 다른 번들들(214)은 동적으로 OSGi 프레임워크(220)에 다운로드 되고 API (Application Program Interface) 콜(call)을 통하여 프레임워크와 통신하며 서비스들과는 API 콜과 미리 정의된 인터페이스 (Interfaces)를 통하여 통신 한다.

[0044] 또한 시스템 번들(213)은 테스터에게 OSGi 프레임워크(220)와 프레임워크에 로드된 번들들을 제어하는 Shell UI 를 제공한다.

[0045] 상기 OSGi 프레임워크(220)는 테스트 케이스 번들(300)들이 로드(load)되고 테스트를 수행할 수 있도록 테스트 실행 환경을 제공한다.

[0046] 상기 어댑터 번들(222)은 테스트 관리 호스트(30)에 존재하면서 OSGi 프레임워크(220)를 테스트 실행 환경의 프레임워크로 사용할 수 있게 하는 기능을 담당한다.

[0047] 상기 디렉터 번들(212)은 테스트 툴의 UI(User Interface)와 테스트를 관리할 수 있는 서비스 및 테스트 툴에서 필요로 되어지는 OSGi 서비스들을 제공하는데 도2c와 도2d 에서 도시한 디렉터 번들 및 서포트 번들이 포함하는 번들들을 포함시켜 구조의 복잡성을 개선할 수 있도록 구성된다.

[0048] 보다 상세하게 상기 디렉터 번들(212)은 도5에서 도시한바와 같이, 디렉터(212) 패키지의 메인(Main) 패키지로 타겟 호스트(10)의 타겟으로부터 뿌려지는 통보 메시지를 처리해서 타겟 호스트(10)의 타겟 등록을 관리하고 타겟 호스트(10)로부터 전송되는 네트워크 메시지를 처리하는 한편 테스트를 수행하여 결과를 처리하는 역할을 하는 코어(Core) 패키지와 GUI의 이벤트와 사용자로부터의 입력을 처리해서 디렉터(212)에게 전달하는 애플릿(Applet) 패키지와 XML(eXtensible Markup Language) 스크립트(Script)로 작성된 테스트 케이스를 처리하는 스

스크립트 패키지와 디렉터(212) 패키지에게 네트워크 처리 모듈과 테스트 케이스 번들(300)에 대한 처리, 테스트 로그 처리 및 테스트 수행에 관련된 유틸리티 API를 제공하는 서비스 패키지와 서비스 패키지의 인터페이스를 가지고 있는 Shared 패키지와 콘솔(Console) 창의 커맨드 라인(Command Line)에서 입력 받은 것을 처리할 수 있게 커맨드(Command) 서비스 패키지와 타겟 호스트(10)의 타겟을 관리하는 타겟 서비스 패키지와 GUI를 제공하는 애플릿 패키지와 XML을 파싱(Parsing)하고 처리하는 XML Parser & Transform 패키지로 이루어지는 코어 패키지 그룹과 코어 패키지 그룹에 필요한 서비스를 제공하는 OSGi 서비스 번들들 코어 패키지 그룹과 OSGi 서비스 번들들을 OSGi 프레임워크(220)에 설치하는 그룹(Group) 패키지로 이루어진다.

- [0049] 이하, 상기 디렉터번들(212)의 핵심 패키지인 `com.samsung.miracle.core.test.director` 의 클래스 다이어그램을 설명한다.
- [0050] 상기 패키지는 도6에서 도시한 바와 같이, 디렉터(212) 내부에 있는 코어(core) 번들의 BundleActivator인 디렉터 클래스와 타겟 호스트(10)의 OSGi 타겟 프레임워크 에서 로드되어 동작하는 에이전트 번들(210)의 브로드캐스팅(broadcasting) 정보를 감지하는 역할을 하는 탐지(Discovery) 클래스와 테스트 툴의 GUI를 담당하는 테스트 애플릿(TestApplet)의 더미 클래스와 디렉터 패키지 내부의 클래스 오브젝트(object)들의 매개체 역할을 하는 핸들러 클래스와 HTTP(Hypertext Transfer Protocol) 서비스에 대한 서비스 이벤트의 트랙커(tracker) 서비스를 담당하는 Http 트랙커 클래스와 타겟 호스트(10)에 위치한 에이전트 번들(210)과 타겟 OSGi 프레임워크(102)의 정보와 동작을 관리하는 RemoteServiceImpl 클래스 및 테스트를 수행하는 Run 클래스로 구성된다.
- [0051] 본 발명의 일실시예에 따른 OSGi 서비스 플랫폼 테스트 툴의 타겟 호스트(10)에서의 구성은 도 7에서 도시한 바와 같이, 에이전트 번들(210)에 의해 구성된다.
- [0052] 타겟 OSGi 프레임워크와 타겟 번들들은 타겟 호스트(10)의 외부 컴포넌트가 되는데, 타겟 번들들은 동적으로 타겟 OSGi 프레임워크에 다운로드 되고 API 콜을 통하여 통신하며 서비스들과는 API 콜과 미리 정의된 인터페이스(Interfaces)를 통하여 통신 한다.
- [0053] 상기 에이전트 번들(210)은 타겟 OSGi 프레임워크(102) 위에 적재되어 테스트 관리 호스트(30)와 타겟 OSGi 프레임워크(102) 사이에서 테스트 케이스 번들(300)과 메시지 이동의 매개체 역할을 한다.
- [0054] 보다 상세하게 에이전트 번들(210)은 도8에서 도시한바와 같이, 타겟 호스트(10)의 위치와 정보를 알리는 통보(Notify)메시지를 뿌리고 테스트 관리 호스트(30)에 위치한 디렉터 (212, Director)번들과 연결 된 후 디렉터 번들(212)의 정보를 관리하고, 디렉터 번들(212)로부터 받은 네트워크 메시지를 처리하며, 디렉터 번들(212)로부터 받은 테스트 케이스 번들(300)을 타겟 호스트(10)의 타겟 OSGi 프레임워크(102)에 설치하는 역할을 하는 에이전트 코어 패키지와 에이전트 코어 패키지에게 네트워크 처리 모듈과 테스트 케이스 번들(300)에 대한 처리와 같은 유틸리티(Utility) API(Application Programming Interface)를 제공하는 서비스(Service) 패키지와 서비스 패키지의 인터페이스(Interface)를 가지고 있는 Shared 패키지와 OSGi.org 프레임워크에서 발생하는 서비스 이벤트(Event)를 처리해서 에이전트 코어 서비스에게 알려 주는 트랙커(Tracker) 서비스로 이루어지는 에이전트코어 패키지 그룹과 에이전트 코어 패키지 그룹에 필요한 서비스를 제공하는 OSGi 서비스 번들들과 에이전트 코어 패키지 그룹과 OSGi 서비스 번들들을 타겟 OSGi 프레임워크에 설치하는 그룹(Group) 패키지를 포함한다.
- [0055] 여기서 도9를통해 상기 에이전트번들(210)의 핵심 패키지인 `com.samsung.miracle.agentcore.test.target` 의 클래스 다이어그램을 설명한다.
- [0056] 상기 패키지는 자신의 존재를 알려 주기 위해 일정한 간격으로 브로드캐스팅(broadcasting)을 하는 스레드(thread)를 실행시키고, 테스트 관리 호스트(30)에서 동작하는 디렉터 번들(212)로부터 받은 테스트 케이스 번들(300)을 타겟 호스트(10)에 위치한 타겟 OSGi 프레임워크에 설치시키며 테스트가 완료되면 테스트 로그와 그 결과를 테스트 관리 호스트(30)에 위치한 디렉터 번들(212)에게 전달하면서 테스트 케이스 번들(300)을 타겟 OSGi 프레임워크(102)에서 제거(uninstall)시키고, 디렉터 번들(212)로부터 전달되는 명령들을 타겟 OSGi 프레임워크에 전달하는 기능을 수행한다.
- [0057] 상기 에이전트 번들(210)의 패키지 번들은 도2a와 도2b 에서 도시한 osgi-refimpl 번들, 타겟 번들이 포함하는 번들들을 포함시켜 구조의 복잡성을 개선할 수 있도록 구성된다. 다만, osgi-refimpl 번들이 포함한 타겟 OSGi 서비스 번들은 제외하고 구성된다. 이때, 제외된 타겟 OSGi 서비스 번들 중에서 테스트에 필요한 타겟 OSGi 서비스 번들은 테스터(tester)가 UI 또는 스크립트를 통해 입력할 수 있다.
- [0058] 결과적으로 상기 에이전트 번들(210)을 통해 테스트 대상이 되지 않는 OSGi 서비스 번들을 설치하지 않아 메모

리를 절약할 수 있다. 예를들어 실제 테스트 대상이 되는 서비스 번들은 6개에 불과하지만 refimpl 번들에 28개의 서비스 번들이 포함되어 있는 경우 종래의 OSGi 테스트 툴에 의하면 28개의 서비스 번들을 모두 일괄적으로 설치하여야 하지만, 본발명의 일실시예에 따르면 6개의 서비스 번들만 독립적으로 설치할 수 있어 메모리를 절약할 수 있다.

- [0059] 상기 본발명의 일실시예에 따른 OSGi 서비스 번들의 설치 및 변경에 대한 과정은 도 18a와 도18b를 통해 종래기술과 비교하여 후술한다.
- [0060] 이하, 상기 테스트 툴을 테스트 작업 수행 과정에 관한 논리적 단위인 모듈단위로 구분하여 설명한다.
- [0061] 도 10은 타겟 호스트(10)의 모듈 다이어그램을 나타낸 것으로, 타겟 호스트(10)의 모듈은 디렉터(212)에게 타겟의 정보를 보내고 테스트가 진행될 때 디렉터(212)로부터 받은 테스트 케이스 번들(300)을 타겟 OSGi.org 프레임워크에 설치하고 테스트 케이스 번들(300)의 로그와 결과를 디렉터(212)에게 전달하는 테스트 케이스 처리부 모듈과 테스트 케이스 처리부에서 필요한 유틸리티 서비스모듈들로 이루어지고, 상기 두 모듈은 하나의 번들로 존재한다.
- [0062] 도 11은 테스트 관리 호스트(30)의 모듈 다이어그램을 나타낸 것으로, 테스트 관리 호스트(30)의 모듈은 테스트 수행 로직 모듈과 GUI를 처리하는 디스플레이 로직 모듈과 XML 페이지를 구문분석(Parsing)해서 처리하는 "XML Parser & Transform 로직 그룹과 테스트 수행 로직에 필요한 유틸리티 로직이 하나의 번들로 합쳐져서 존재한다.
- [0063] 도 12는 타겟 호스트(10)에 위치한 에이전트 번들(210)의 세부 모듈을 나타낸 것으로, 에이전트 번들(210)은 에이전트 코어 모듈(242)과 네트워크 디렉터 패킷 처리 모듈(245)과 콘솔 커맨드 처리 모듈(241)과 콘솔 모듈(240)로 구성된다.
- [0064] 상기 에이전트 코어 모듈(242)은 타겟 호스트(10)의 위치와 정보를 알리는 통보 메시지를 뿌리고, OSGi 번들로서의 기능을 하며 테스트 관리 호스트(30)의 디렉터 번들(212)을 관리하는 디렉터 관리 모듈(243)과 테스트가 진행될 때 디렉터 번들(212)로부터 받은 테스트 케이스 번들(300)을 타겟 OSGi 프레임워크에 설치하고 테스트 케이스 번들(300)로부터 받은 로그와 테스트 결과를 디렉터 번들(212)에게 전달하는 기능을 수행하는 테스트 케이스 관리 모듈(244)을 포함한다.
- [0065] 상기 네트워크 패킷 처리 모듈(245)은 네트워크로 받은 패킷을 처리해서 에이전트 코어 모듈(242)로 전달해준다.
- [0066] 콘솔 커맨드 처리 모듈(241)은 테스트 케이스 수행중 발생하는 로그나 에이전트 코어(242)에서 발생하는 로그를 콘솔(Console) 윈도우(window)에 출력해준다.
- [0067] 도 13은 테스트 관리 호스트(30)에 위치한 디렉터 번들(212)의 세부 모듈을 나타낸 것으로, 핸들러 모듈(255)과 테스트 컨트롤 모듈(256)과 네트워크 패킷 처리 모듈(258)과 타겟 탐지 모듈(258)과 OSGi 번들 라이프사이클 관리 모듈(253)과 GUI 처리 모듈(252)과 콘솔 커맨드 처리 모듈(251)로 이루어진다.
- [0068] 상기 핸들러 모듈(255)은 디렉터 번들(212) 내부 모듈 간의 메시지 전달을 담당하는 모듈 간 통신 관리 모듈과 OSGi 프레임워크에서 발생하는 서비스 이벤트를 감지(detect)해서 해당하는 모듈에 이벤트를 전달하는 OSGi 이벤트 트래커 모듈로 구성된다.
- [0069] 상기 테스트 컨트롤 모듈(256)은 테스트의 요구에 의해서 테스트가 진행되면 전체 테스트 수행을 관장하는 테스트 수행 모듈과 테스트 케이스 번들(300)이 위치하는 폴더를 관리해서 테스트의 요구에 따라서 원하는 폴더에서 테스트 케이스 번들(300)들을 OSGi 프레임워크에 설치하는 테스트 케이스 폴더 관리 모듈과 테스트 케이스 번들(300)이 사용하는 Http 서버를 관리하는 Http 서버 관리 모듈과 테스트 수행 중 에이전트 번들(210)로부터 받은 로그를 처리하고 디렉터 번들(212)의 GUI에 출력되는 로그들을 텍스트 형식의 파일로 저장하는 로그 처리 모듈로 구성된다.
- [0070] 상기 네트워크 패킷 처리 모듈(259)은 에이전트 번들(210)로부터 받은 패킷을 처리해서 핸들러 모듈(255)과 테스트 컨트롤 모듈(256)에 전달하고 상기 두 모듈에서 에이전트 번들(210)로 전달하려는 패킷을 받아서 처리한다.
- [0071] 상기 타겟 탐지 모듈(258)은 디렉터 번들(212)이 실행되는 동안 에이전트 번들(210)이 브로드캐스팅으로 뿌리는 통지 패킷을 감지하여 감지된 타겟 호스트(10)의 정보가 있으면 타겟 관리 모듈(254)에 전달하여 처리하게

한다.

- [0072] 상기 OSGi 번들 라이프사이클 관리 모듈(253)은 테스터의 요구에 따라 OSGi 프레임워크(220)에 설치된 번들의 라이프사이클(Lifecycle) 정보를 제공하고 번들의 상태(State)를 변경하거나 새로운 번들을 OSGi 프레임워크에 설치할 수 있는 기능을 제공한다.
- [0073] 상기 GUI 처리 모듈(252)은 GUI를 통해서 발생하는 테스터의 요구를 처리하여 핸들러 모듈(255)에게 전달한다.
- [0074] 상기 콘솔 커맨드 처리 모듈(251)은 콘솔 창에서 입력되는 테스터의 커맨드 명령을 처리하는데, 이때 콘솔 윈도우를 제공하는 UI와 테스트 툴의 GUI의 동기(synchronization)가 맞도록 구성하여 테스트 중에 오류가 발생할 때, OSGi 프레임워크나 서비스번들의 상태 확인과 오류 원인 분석이 가능하도록 구현할 수 있다. 예를 들어, 콘솔 창 등에 오류에 대응할 수 있는 프롬프트 등에 의한 커맨드 명령 입력 수단을 제공할 수 있도록 구현하여 테스트 중에 발생한 오류에 대응하고, 오류의 원인 분석을 가능하게 한다.
- [0075] 도 14는 본 발명의 일실시예에 따른 OSGi 테스트 진행 절차를 나타내는 도면이다.
- [0076] 도 14에서 도시한바와 같이, 본 발명의 일실시예에 따른 OSGi 서비스 플랫폼을 테스트 방법은 디렉터 번들을 테스트 관리 호스트의 프레임워크에 설치하고, 에이전트 번들과 테스트할 서비스 번들을 타겟 호스트의 프레임워크에 설치하는 단계(S1)와 테스트 케이스 번들을 설치하여 테스트를 수행하는 단계(S2) 및 상기 테스트의 결과 및 테스트 로그를 제공하는 단계(S3)를 포함한다.
- [0077] 우선, OSGi 서비스 플랫폼에서 서비스를 제공하는 기능적 배포 단위인 번들(Bundle)의 상태(state) 변화에 대해 설명한다.
- [0078] 도 15에서 도시한바와 같이, 번들의 라이프-사이클(Life-Cycle)은 번들이 OSGi 프레임워크에 설치되어 제거될 때까지 외부의 정의되어진 동작 명령에 의해 변화(Explicit Transition)하거나, 정의되어진 조건에 만족되면 자동적으로 진행(Automatic Transition)한다.
- [0079] OSGi 프레임워크는 OSGi 서비스 플랫폼의 핵심 구성요소로서 번들을 동적으로 설치(INSTALL), 갱신(UPDATE)할 수 있는데, 번들이 OSGi 프레임워크에 설치되어 갱신 또는 재시작되거나, 제거될 때까지 OSGi 프레임워크는 번들 상태를 구분하여 라이프-사이클을 관리하는데 그 상태는 번들이 성공적으로 설치된 상태를 나타내는 "INSTALLED"와 번들이 필요로 하는 모든 Java 클래스 파일들과 Native code가 OSGi 프레임워크 실행환경에 설치된 상태를 나타내는 "RESOLVED", 번들이 실행되고 있는 상태를 나타내는 "STARTING", 번들이 중지되고 있는 상태를 나타내는 "STOPPING", 번들이 성공적으로 실행되고 동작하고 있는 상태를 나타내는 "ACTIVE", 마지막으로 번들이 제거된 상태를 나타내는 "UNINSTALLED"로 나눌 수 있다.
- [0080] 상기 디렉터 번들과 에이전트 번들 및 테스트할 서비스 번들을 설치하는 단계(S1)는 테스트 관리 호스트(30)에서 OSGi 프레임워크(220)가 자바 가상 머신(Java Virtual Machine) 위에 설치(S10)되고 그 위에 디렉터 번들(212)이 설치(S10)되는 단계와 상기 단계(S10)와 동시에 타겟 호스트(10)에서 타겟 OSGi 프레임워크가 자바 가상 머신 위에 설치되고 그 위에 에이전트 번들(210)과 테스트 대상이 되는 OSGi 서비스 번들이 설치되는 단계(S12)를 포함한다.
- [0081] 상기 OSGi 서비스 번들은 테스터가 스크립트 또는 UI를 통해 입력할 수 있는데 스크립트를 통한 입력은 osgi.properties 파일에 테스트 대상 OSGi 서비스 번들을 추가시키는 방식이 사용될 수 있다. 테스트에 필요한 OSGi 서비스 번들들의 개발등에 의해 변경이 필요한 경우에 대해서 도18a와 도18b를 통해 종래 방식과의 차이를 설명한다.
- [0082] 상기 테스트 케이스 번들을 설치하여 테스트를 수행하는 단계(S2)는 우선, 디렉터 번들(212)이 테스트 툴의 GUI(Graphic User Interface)를 띄우면 에이전트 번들(210)이 브로드캐스팅을 이용해 타겟호스트에 대한 위치를 알리는 통보 메시지(message)를 뿌린다.(S20)
- [0083] 다음으로 디렉터 번들(212)이 에이전트 번들(210)로부터 받은 통보 메시지를 분석하여 테스트 툴의 GUI에 타겟 호스트(10) 정보를 디스플레이(S22)하면 테스터(tester)는 테스트 관리 호스트(30)에 위치한 폴더에서 테스트 케이스 번들(300)을 OSGi 프레임워크에 설치하고(S24), 테스트할 타겟 호스트(10)를 선택한 후, 테스트 툴 GUI의 시작버튼을 누르면 테스트가 자동으로 시작된다.(S26)
- [0084] 상기 테스트 케이스 번들은 테스터가 스크립트 또는 UI를 통해 입력할 수 있는데, 스크립트를 통한 테스트 케이스 번들의 입력은 테스트 케이스 번들들을 모아놓은 파일(osgi.properties 파일 등)에 테스트 케이스 번들을

추가시키는 방법으로 구현할 수 있다. 또한 콘솔창이 제공하는 커맨드(command) 명령을 통해서도 테스트 케이스 번들의 입력이 가능하다.

- [0085] 본발명에 의한 테스트 케이스 번들의 추가는 테스트가 진행되기 전 뿐만 아니라 테스트 진행중에도 테스트 과정의 종료없이 가능하고, 테스트 케이스 번들별로 독립적으로 추가 시킬수 있어 기존의 테스트 케이스 폴더를 통해 집합적으로 관리하던 방식에 비해 테스트 케이스 번들 추가가 쉽고 실시가 용이한 장점이 있다.
- [0086] 이하, 도16a와 도16b를 통해 종래기술과 본발명의 테스트툴에서 테스트 케이스 번들들을 추가할 때의 차이를 테스트 관리 호스트(30)에 위치한 모듈들의 스택(Stack) 구조를 통해 설명한다.
- [0087] 도 16a에서 도시한 바와 같이, 실행 패치(Batch) 파일에 의해서 자동으로 OSGi.org 프레임워크가 실행되어 그 위에 테스트 관리 호스트(30) 쪽의 OSGi.org 테스트 툴이 필요한 유틸리티 모듈이 적재 되면 그 것을 기반으로 해서 테스트 수행 모듈이 실행 되어 테스트 케이스들의 집합인 테스트 케이스 번들들이 적재되는 방식으로 동작한다.
- [0088] 이때, 새로운 테스트 케이스 번들의 개발 등에 의해 추가, 수정 등의 테스트 케이스 번들을 변경 작업을 하는 경우 압축된 파일을 풀고 변경 사항을 반영(변경된 파일 내용 수정)한 후 다시 압축과정을 진행하는 작업이 필요하다.
- [0089] 반면, 도 16b는 본 발명에 의한 테스트 툴에 의하면, 실행 패치 파일에 의해서 자동으로 OSGi 프레임워크(220)가 실행되어 그 위에 테스트 관리 호스트(30) 쪽의 테스트 툴이 필요한 유틸리티 모듈이 적재 되면 그것을 기반으로 해서 테스트 수행 모듈이 실행되고 테스터(tester)가 원하는 시점에서 전술한 바와 같이, UI 등을 통해 원하는 폴더에 위치한 하나 이상의 테스트 케이스 번들(300)들을 입력하여 동적으로 OSGi 프레임워크(220)에 적재할 수 있다.
- [0090] 따라서, 종래의 OSGi 테스트 툴과 달리 테스트 케이스 번들이 집합을 이루어 프레임워크에 자동적으로 적재되지 않고 테스터에 의해 동적으로 설치되므로, 테스트 케이스의 변경은 테스터가 단지 변경을 원하는 테스트 케이스 번들을 선택하는 방법으로 구현이 가능하다. 여기서, 테스터가 테스트 케이스 번들을 동적으로 선택하는 과정은 윈도우 대화창 등과 같은 UI 화면으로 구현하는 것이 바람직하고 도19를 통해 후술한다. 상기 테스트 케이스의 변경하거나 새로 개발한 테스트 케이스 번들의 추가는 도 18a와 도18b를 통해 종래기술과 비교하여 후술한다.
- [0091] 테스트가 시작되면 디렉터 번들(212)은 테스터에 의해 선택되어진 테스트 케이스 번들(300)들 중에서 가장 상위로 선택된 테스트 케이스 번들(300) 하나를 네트워크를 통해서 타겟 호스트(10)의 에이전트 번들(210)에게 전달하고(S28), 테스트 케이스 번들(300)을 받은 에이전트 번들(210)은 이것을 타겟 OSGi 프레임워크에 설치하여 OSGi 서비스 플랫폼의 특정 기능을 검증하는 테스트를 수행한다.(S29)
- [0092] 상기 테스트 결과와 테스트 로그를 출력하는 단계(S3)에서는 우선 첫 단계에서 테스트 케이스 번들(300)이 테스트의 결과 및 로그를 에이전트 번들(210)을 통해서 테스트 관리 호스트(30)에 위치한 디렉터 번들(212)에게 전달하고, 테스트툴의 GUI에 디스플레이 한다(S30). 테스트가 종료 되면 에이전트 번들(210)은 타겟 OSGi 프레임워크에서 해당 테스트 케이스 번들(300)을 제거(S32)하고, 다음 순서의 테스트 케이스의 존재여부를 판단(S34)하여, 존재하면 해당 테스트 케이스에 대해서 다시 상기 테스트를 수행하고, 존재하지 않으면 테스트가 종료된다(S36).
- [0093] 상기 테스트 로그는 각각의 테스트 케이스 번들들에 대하여 테스트가 수행된후 XML(eXtensible Markup Language)을 이용하여 요약 페이지 형식으로 테스트 툴의 GUI를 통해 제공된다. 뿐만 아니라, 본 발명의 일실시예에 따른 상기 테스트 로그는 텍스트 형식의 파일로 제공될 수 있어 테스트 과정 중이 아닌 경우에 있어서도 테스트 로그의 확인을 용이하게 해준다.
- [0094] 전술한 과정에서, 디렉터 번들(212)이 테스트 관리 호스트(30)에 위치한 OSGi 프레임워크(220)에 설치되는 과정과 상기 설치된 디렉터 번들(212)이 동작하는 과정과 상기 디렉터 번들(212)이 번들으로써 중지되는 과정 및 상기 디렉터 번들(212)이 중지 또는 설치된 상태에서 제거되는 과정에 대한 세부 단계는 OSGi 스펙(OSGi Service Platform Release 2 ,October 2001)에 기술되어 있다.
- [0095] 도 17은 본 발명의 일실시예에 따른 테스트 진행과정을 테스트 툴의 구성을 통해 보여주는 도면이다.
- [0096] 도 17에서 도시한 바와 같이, 에이전트 번들(210)이 설치되어 호스트의 IP 주소와 타겟 OSGi 프레임 정보를 알리는 통보 메시지를 뿌리면 프레임워크에 설치된 디렉터 번들(212)이 상기 메시지를 감지하여 테스트 툴의 GUI

에 디스플레이하고, 테스터(tester)가 테스트 케이스 번들(300)을 SWC OSGi 프레임워크(220)에 설치 한 후에 테스트 툴의 GUI를 통해서 시작(start) 버튼을 클릭하면(도 17의 ①), OSGi 프레임워크에 설치된 테스트 케이스 번들(300)이 네트워크를 통해서 에이전트번들(210)에 전달되고, 전달된 테스트 케이스 번들(300)이 타겟 OSGi 프레임워크에 설치되어 테스트를 수행하게 된다(도 17의 ②).

- [0097] 테스트의 결과와 테스트 로그(Log)는 에이전트 번들(210)에 의해 다시 네트워크를 통해서 디렉터 번들(212)에 전달되어 테스트 툴의 UI에 디스플레이 되고 동시에 더 자세한 테스트 로그를 텍스트 파일 형식의 테스트 결과 페이지(Test Summary Page)에 기록하게 된다(도 17의 ③).
- [0098] 도 18a 와 18b는 OSGi 서비스 번들과 테스트 케이스 번들(300)들을 새로 개발 했을 경우 종래기술 또는 본 발명의 일실시예에 따른 테스트 툴을 이용해서 테스트하는 절차를 각각 보여주는 도면이다.
- [0099] 우선 새로 개발한 OSGi 서비스 번들인 번들 A1을 테스트 툴에 추가하고자 할 경우 종래기술에 의한 테스트툴의 경우 도18a에서 도시한 바와 같이, osgi-refimpl 번들의 osgi-reimpl.jar 파일에 압축을 풀어 개발전 번들 A를 제거하고, 개발한 번들인 번들 A1을 추가한 후 압축하여 osgi-reimpl.jar 파일을 생성하는 과정으로 진행한다.
- [0100] 그러나 본발명의 일실시예에 따른 테스트툴의 경우 도18b에서 도시한 바와 같이, 번들 A1을 테스트 툴의 UI 또는 스크립트를 통해 타겟 OSGi 프레임워크의 위치에 복사하기만 하면 된다.
- [0101] 또한, 새로 개발한 테스트 케이스들인 TCA1 부터 TCA10을 테스트 툴에 추가하고자 할 경우 종래기술에 의한 테스트툴의 경우 도18a에서 도시한 바와 같이, osgi-test.jar 파일에 압축을 풀어 새로 개발한 테스트 케이스들인 TCA1 부터 TCA10을 추가한 후 압축하여 osgi-test.jar 파일을 생성하는 과정으로 진행한다.
- [0102] 그러나 본발명의 일실시예에 따른 테스트툴의 경우 도18b에서 도시한 바와 같이, 새로 개발한 테스트 케이스들인 TCA1 부터 TCA10을 테스트 툴의 UI 또는 스크립트를 통해 디렉터 번들(212)의 테스트 케이스 폴더에 추가하기만 하면 된다. 이때, 테스트 케이스의 추가나 변경은 테스트 진행 전 뿐만 아니라 테스트가 진행 중인 경우에도 가능하다. 따라서, 테스터는 원하는 시점에 원하는 테스트 케이스 번들에 따라 테스트 과정을 진행할 수 있으므로 종래기술과 달리 테스트 케이스 별로 테스트 과정을 제어할 수 있게 된다.
- [0103] 도 19는 본 발명의 일실시예에 따른 테스트 관리 호스트(30)의 디렉터 번들(212)에 의한 테스트 툴의 GUI 구성을 보여주는 도면이다.
- [0104] 도 19의 ①번 박스는 타겟 호스트(10)에 위치한 에이전트 번들(210)이 보낸 정보를 테스트 관리 호스트(30)에 위치한 디렉터 번들(212)이 감지하여, 감지한 에이전트 번들(210)의 IP(Internet Protocol)를 표시해주는데, 두 개 이상의 에이전트 번들(210)이 감지 될 경우 콤보 박스를 열어(Drop-down) 모든 에이전트 번들(210)의 IP를 보여 준다. 그래서 테스터는 두 개 이상의 에이전트 번들(210)이 감지되었을 경우 콤보 박스를 열어 특정 호스트를 선택하여 테스트를 수행할 수 있다.
- [0105] 도 19의 ②번 박스는 네트워크에 트래픽(traffic)이 너무 많거나 부하가 걸리는 경우 에이전트 번들(210)이 보낸 정보를 디렉터 번들(212)이 감지 못할 수도 있기 때문에 테스터가 ②번 박스의 IP 텍스트 박스에 직접 에이전트 번들(210)이 존재하는 호스트의 IP를 입력해서 테스트를 수행할 수 있게 해준다.
- [0106] 도 19의 ③번 박스는 디렉터 번들(212)이 감지한 에이전트 번들(210)이 로드된 시스템의 OS(Operating System), OSGi 프레임워크의 제조사 및 자바 가상 머신의 정보를 보여 준다.
- [0107] 도 19의 ④번 박스는 테스트를 수행할 때의 옵션을 선택할 수 있는 테스트 옵션 체크박스(CheckBox)로 디렉터 번들(212)이 위치한 테스트 관리 호스트(30)의 타겟 소프트웨어만 선택해서 테스트를 할 수 있게 설정하는 Local only 옵션과 테스트 과정 중에 문제가 발생하였을 때 문제의 원인을 알 수 있도록 하는 디버그 옵션과 테스트 수행 중에 시간 제한(time out)을 받지 않게 설정하는 No timeouts 옵션과 테스트를 한 단계씩 테스터가 요청할 때 마다 진행하게 설정하는 Single Step 옵션과 테스트를 수행하는 동안 발생하는 로그를 ⑦번 박스의 메시지 윈도우에 모두 보여 주게 설정하는 See all logs 옵션으로 구성된다.
- [0108] 도 19의 ⑤번 박스는 테스트에 관련된 각종 명령을 실행할 수 있는 테스트 메뉴 버튼으로 테스트 관리 호스트(30)에 있어서 테스트 툴의 실행 파일이 존재하는 하위 폴더에 위치하는 테스트 케이스 폴더에 테스트 케이스를 OSGi 프레임워크에 설치(INSTALLED/RESOLED/STARTING/ACTIVE 상태 변화)하는 테스트 케이스 설치(TCs install) 버튼과 테스터가 테스트 케이스 번들(300)과 기타 다른 번들들을 관리(Install/Start/Stop/Uninstall/Update)할 수 있도록 다이얼로그 박스를 실행시키는 메니저 버튼과 테스트 케이스 폴더를 임의의 원하는 폴더로 변경하여 테스트 케이스 번들(300)을 설치시키는 변경 테스트 케이스 디렉토리(Change TC dir) 버튼과 메시지 윈도우

에 있는 로그를 텍스트(.txt 포맷) 파일로 저장하는 텍스트 형식 저장(save text)버튼과 ⑥번 박스에서 선택한 테스트 케이스에 대해서 테스트를 실행하게 하는 시작(start) 버튼으로 구성된다.

- [0109] 상기 변경 테스트 케이스 디렉토리 버튼은 전술한 바와 같이 테스트 과정에서 테스트 케이스를 변경하거나 추가하고자 할 때 이용되는 것으로, 테스트는 테스트 툴의 UI를 통해서 변경 또는 추가하려는 테스트 케이스를 디렉토리에 추가하는 방식으로 구현된다.
- [0110] 도 19의 ⑥번 박스는 상기 ⑤번 박스의 테스트 케이스 설치(TCs install) 버튼을 클릭해서 테스트 관리 호스트(30)의 테스트 케이스 폴더에 위치한 테스트 케이스 번들(300)이 OSGi 프레임워크(220)에 설치된 테스트 케이스 번들(300)을 보여주는 테스트 케이스 리스트 윈도우(Test Case List Window)이고 테스트는 테스트 케이스를 클릭하거나 마우스 드래그를 통해서 원하는 특정 테스트 케이스만을 선택해서 테스트를 실행할 수 있으며 테스트 케이스를 하나도 선택하지 않으면 디폴트(default)로 전체 테스트 케이스가 테스트의 실행 대상이 되어 테스트를 실행할 수 있다.
- [0111] 도 19의 ⑦번 박스는 테스트가 실행되는 동안 발생하는 테스트의 로그와 XML 형식의 테스트 결과 페이지(Test Summary Page)의 URL을 보여주는 메시지 윈도우이고 마지막으로 도 18의 ⑧번 박스는 현재 진행 중인 테스트의 진행 정보를 보여주는 테스트 진행바(Progress bar)이다.
- [0112] 도 20은 도 19의 ⑤번 박스에 있는 변경 테스트 케이스 디렉토리(change TC dir) 버튼을 클릭했을 때 생성되는 다이얼로그(dialog)로 이것을 통해서 테스트는 테스트 관리 호스트(30)에 존재하는 테스트 케이스 폴더를 임의의 다른 폴더로 선택할 수 있고, 선택 되어진 폴더는 테스트 케이스 번들(300)이 위치하는 폴더로 지정된다.
- [0113] 도 21은 테스트가 수행된 후 테스트 결과와 로그에 대한 상세한 정보를 알 수 있는 XML 포맷의 테스트 요약 내용을 보여주는 도면이다.
- [0114] 도 21에서 도시한 바와 같이, 테스트가 완료 되었을 때 도 19의 ⑦번 박스의 Message Window에 표시되는 URL을 인터넷 브라우저(ex. Internet Explorer 또는 Netscape Navigator 등)에서 검색하면 왼쪽 상단의 그림처럼 테스트 결과(Test Result) 웹 페이지 로딩/loading)되고, 상기 웹 페이지의 FORMATTED를 클릭하면 오른쪽 하단의 그림처럼 XML 포맷(format)의 테스트 요약 내용이 로딩된다. 테스트 요약 내용은 실행을 완료한 테스트 케이스의 테스트 Pass/Fail 결과와 테스트 실행 중 발생한 로그 및 타겟 OSGi 프레임워크와 타겟 호스트(10) OS의 정보를 상세하게 보여 준다. 상기 테스트 로그는 본 발명의 이점 및 특징, 그리고 그것들을 달성하는 방법은 첨부되는 도면과 함께 상세하게 후술되어 있는 실시예들을 참조하면 명확해질 것이다. 그러나 본 발명은 이하에서 개시되는 실시예들에 한정되는 것이 아니라 서로 다른 다양한 형태로 구현될 것이며, 단지 본 실시예들은 본 발명의 개시가 완전하도록 하며, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 발명의 범주를 완전하게 알려주기 위해 제공되는 것이며, 본 발명은 청구항의 범주에 의해 정의될 뿐이다. 명세서 전체에 걸쳐 동일 참조 부호는 동일 구성 요소를 지칭한다.

발명의 효과

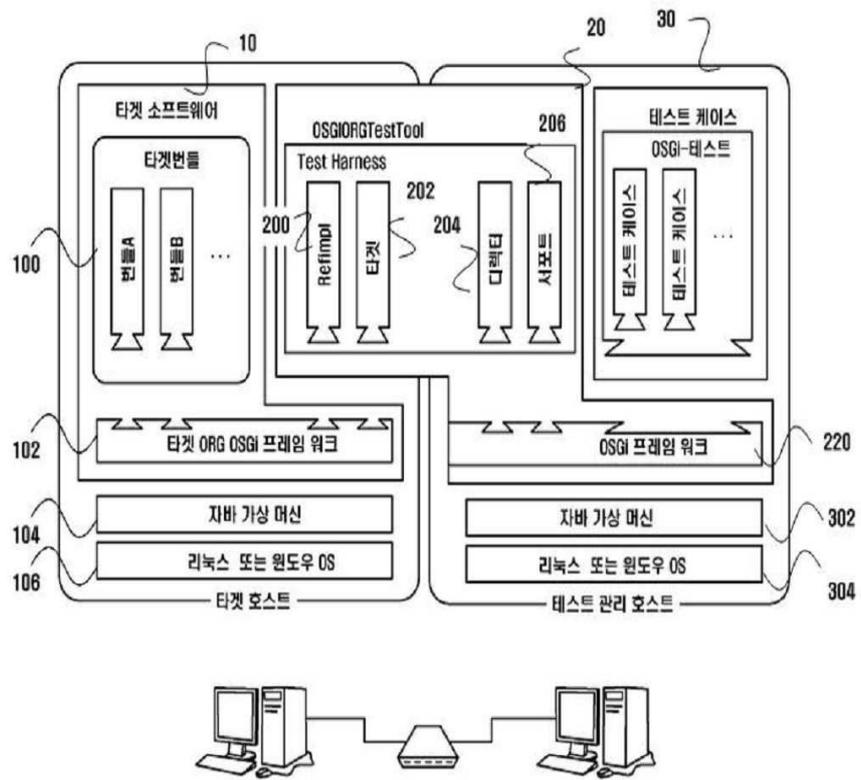
- [0115] 상기한 바와 같은 본 발명의 OSGi 서비스 플랫폼을 테스트 방법 및 이를 이용한 테스트 툴에 따르면 다음과 같은 효과가 하나 혹은 그 이상 있다.
- [0116] 첫째, OSGi 테스트 툴의 번들을 통합하여 구조의 복잡성을 개선할 수 있는 장점이 있다.
- [0117] 둘째, 테스트 케이스나 테스트할 OSGi 서비스 번들의 추가나 변경을 편리하게 할 수 있는 장점도 있다.
- [0118] 셋째, 테스트 과정에서 오류가 발생하는 경우 오류 원인을 용이하게 분석할 수 있는 장점도 있다.
- [0119] 넷째, 테스트 로그(Log)를 텍스트 형식의 파일로 제공하여 테스트 과정외의 시점에서 테스트 로그의 분석을 용이하게 할 수 있는 장점도 있다.

도면의 간단한 설명

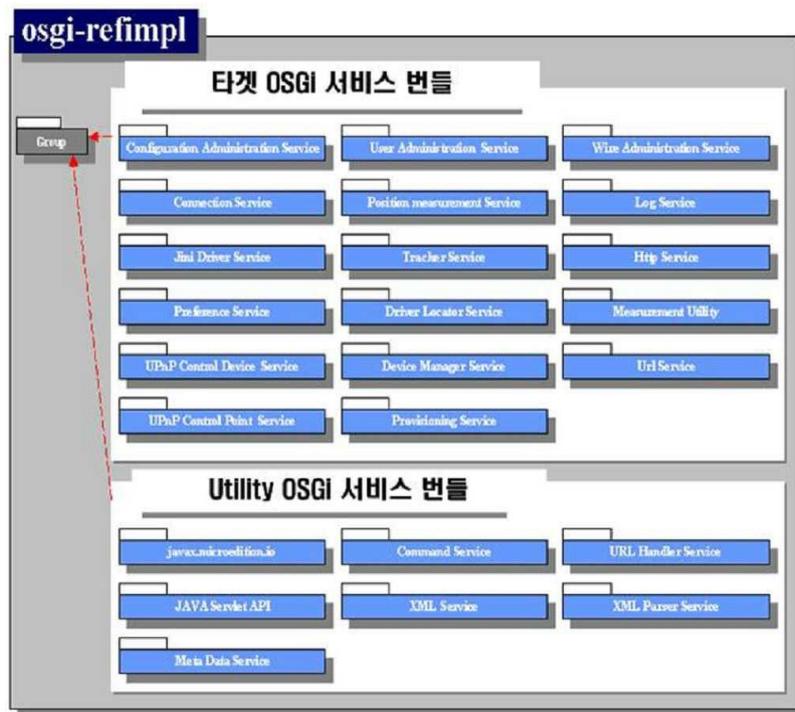
- [0001] 도 1은 종래기술에 의한 OSGi.org에서 제공하고 있는 OSGi 서비스 플랫폼 테스트 툴의 전체 구조를 보여주는 도면이다.
- [0002] 도 2a 내지 2d는 각각 종래기술에 의한 OSGi 서비스 플랫폼 테스트 툴의 osgi-refimpl 번들, 타겟 번들, 디렉터 번들 및 서포트 번들의 패키지 다이어그램을 보여주는 도면이다.

도면

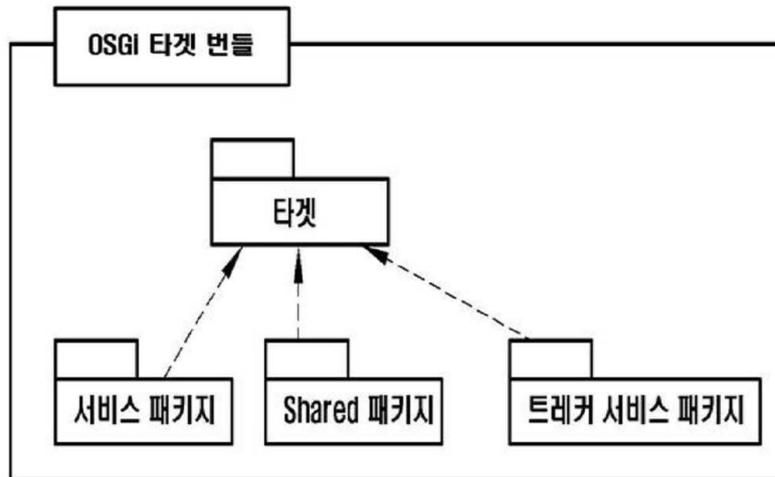
도면1



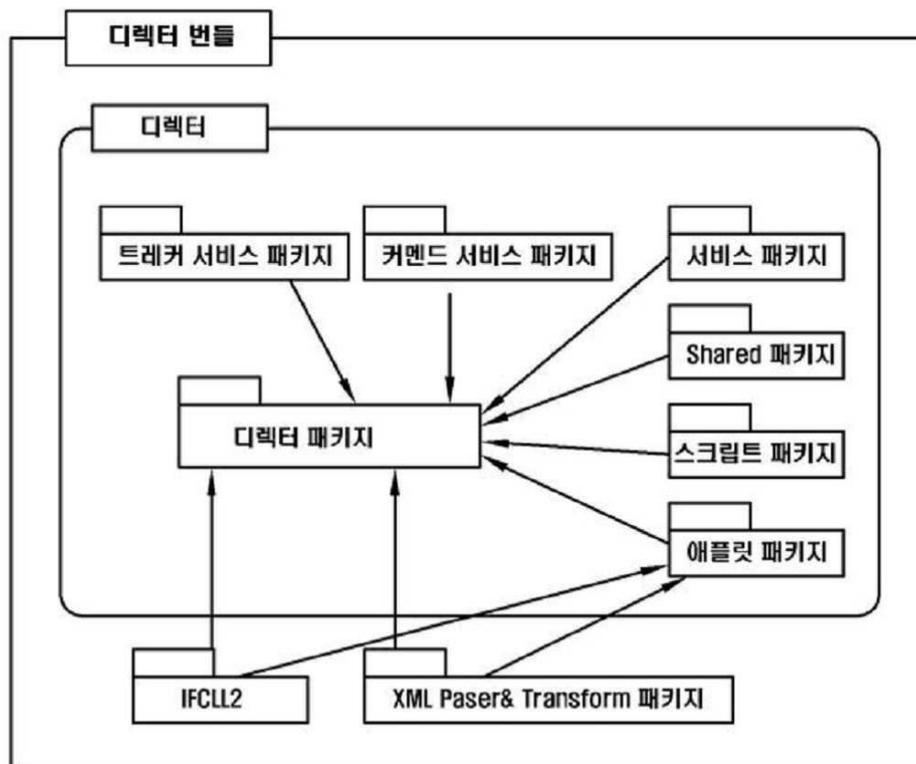
도면2a



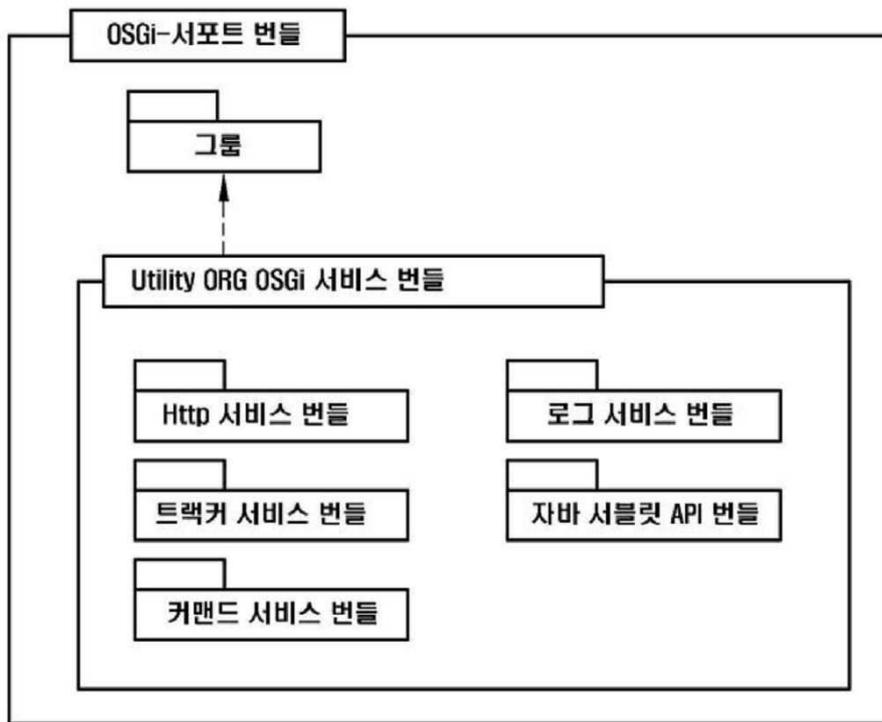
도면2b



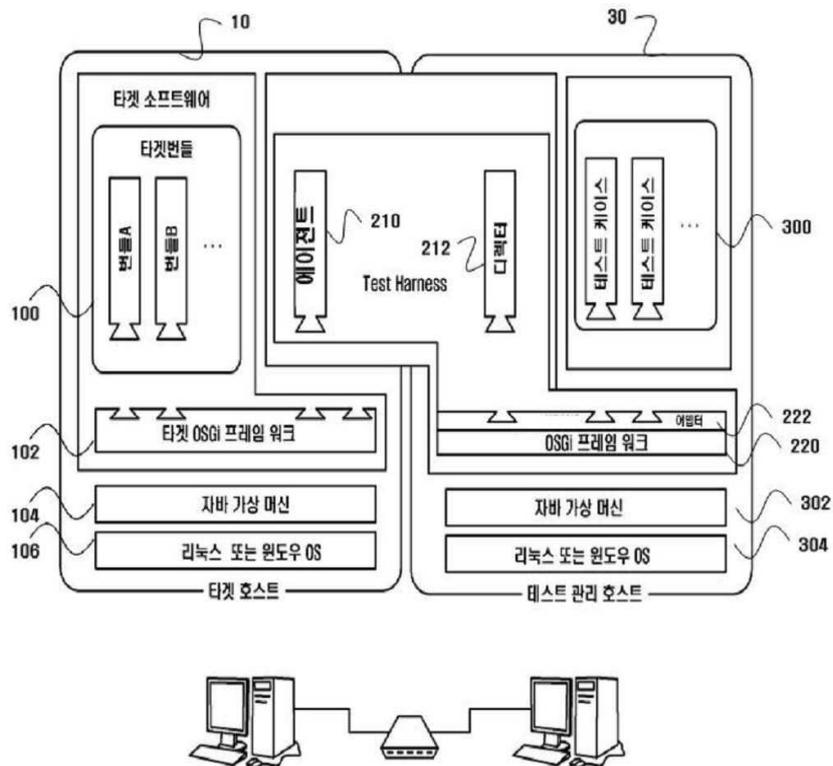
도면2c



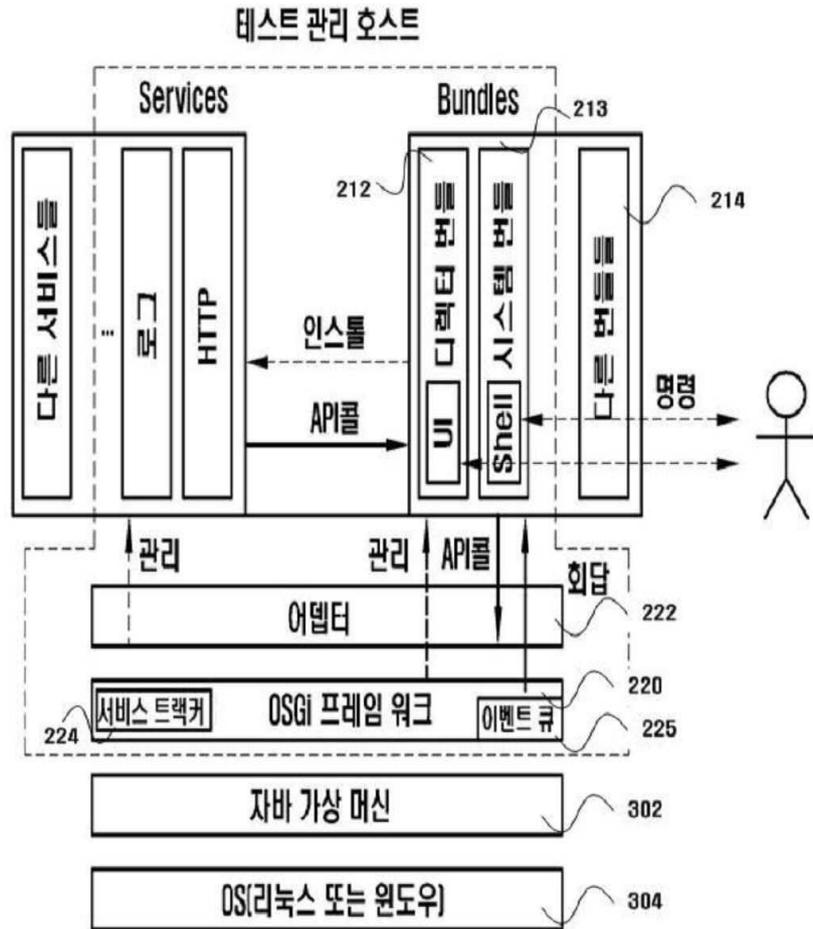
도면2d



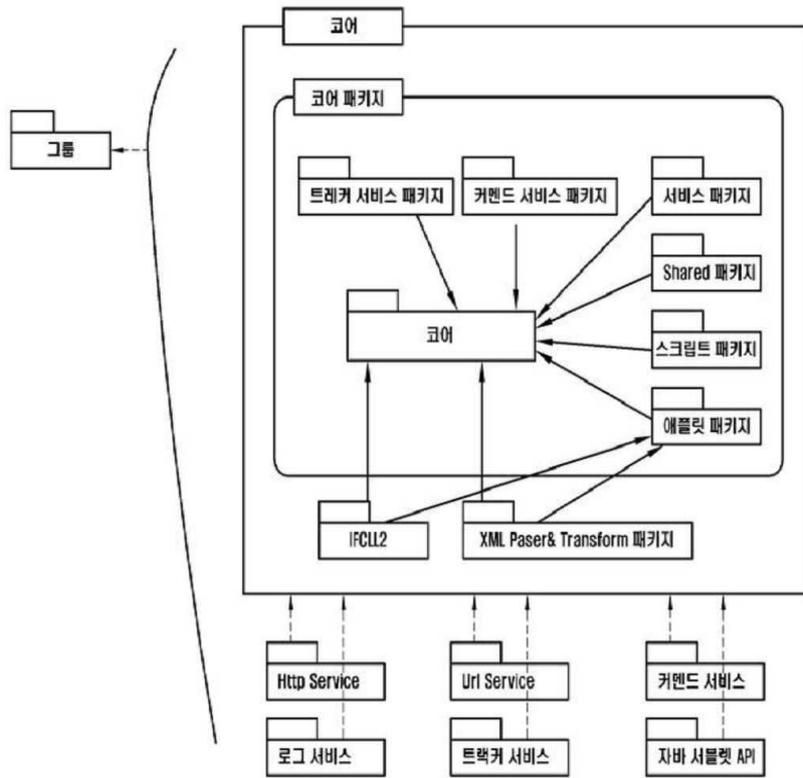
도면3



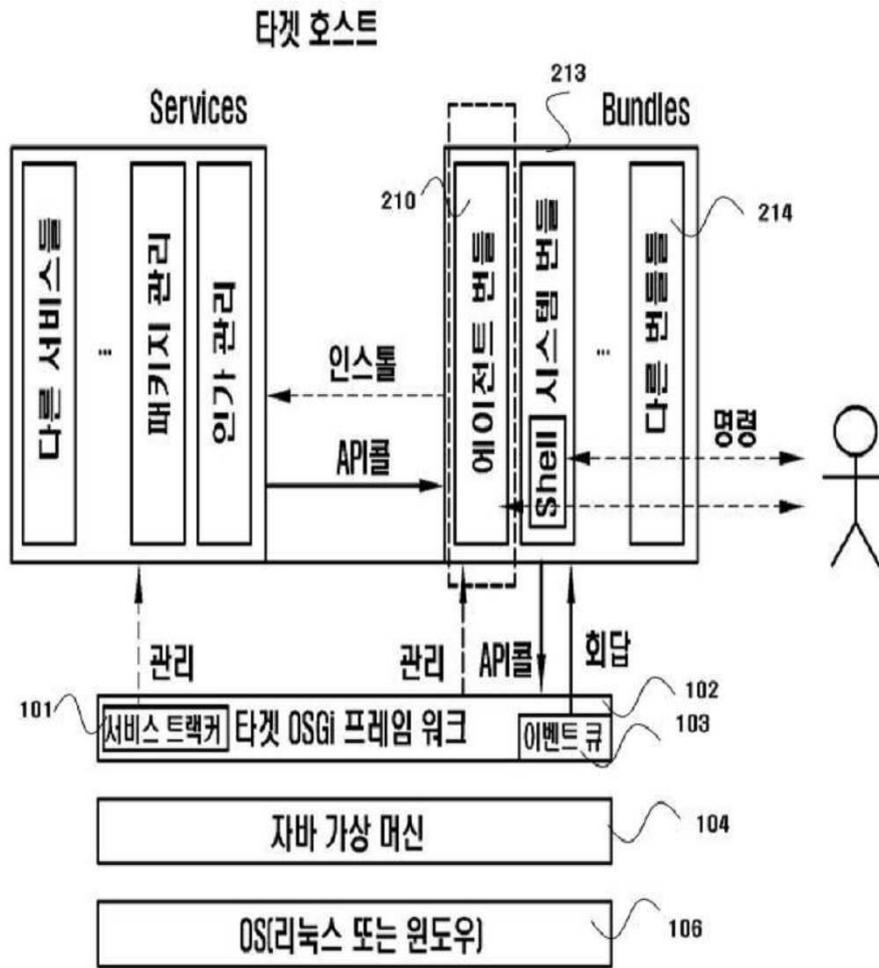
도면4



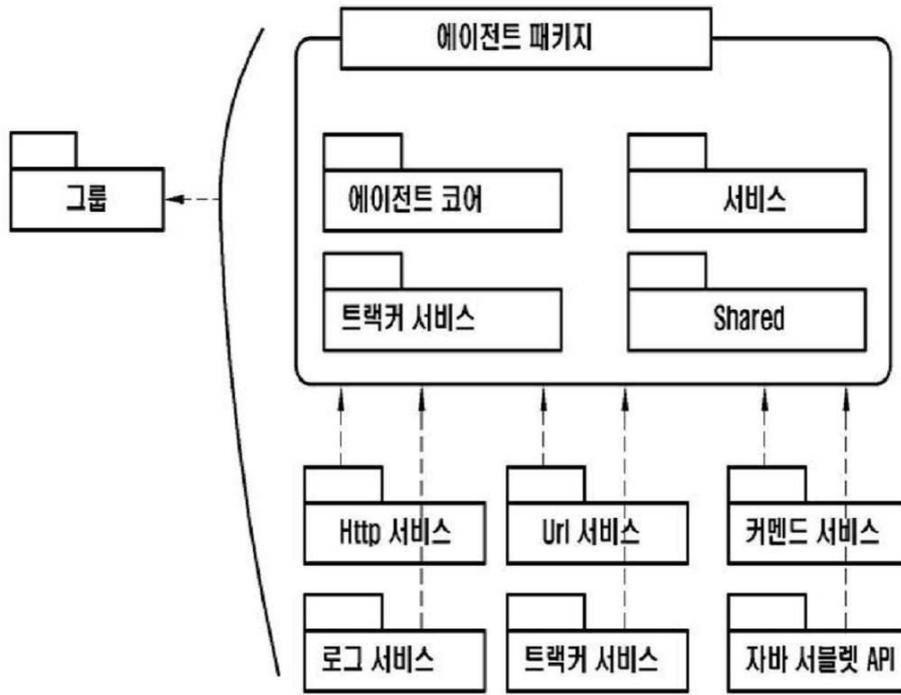
도면5



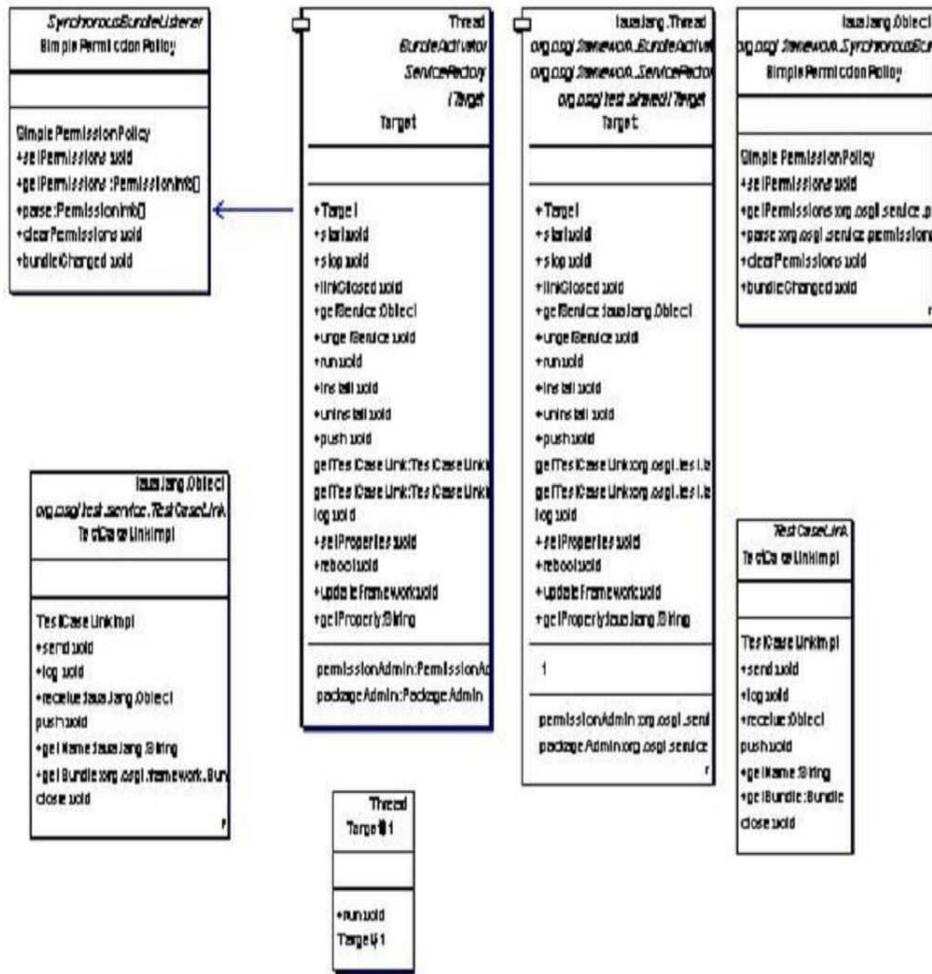
도면7



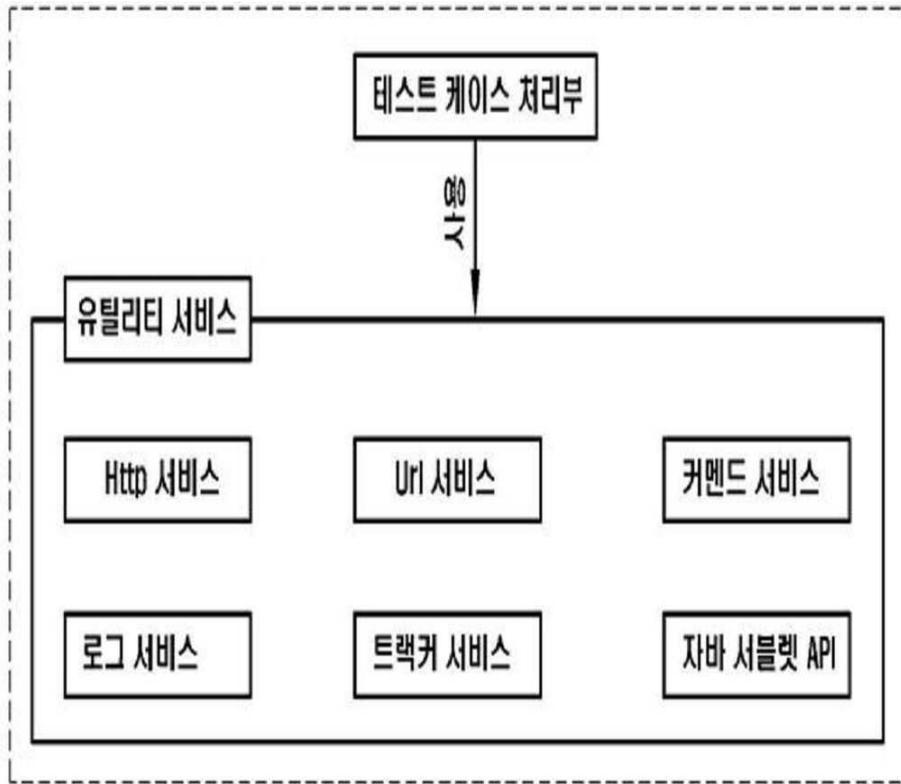
도면8



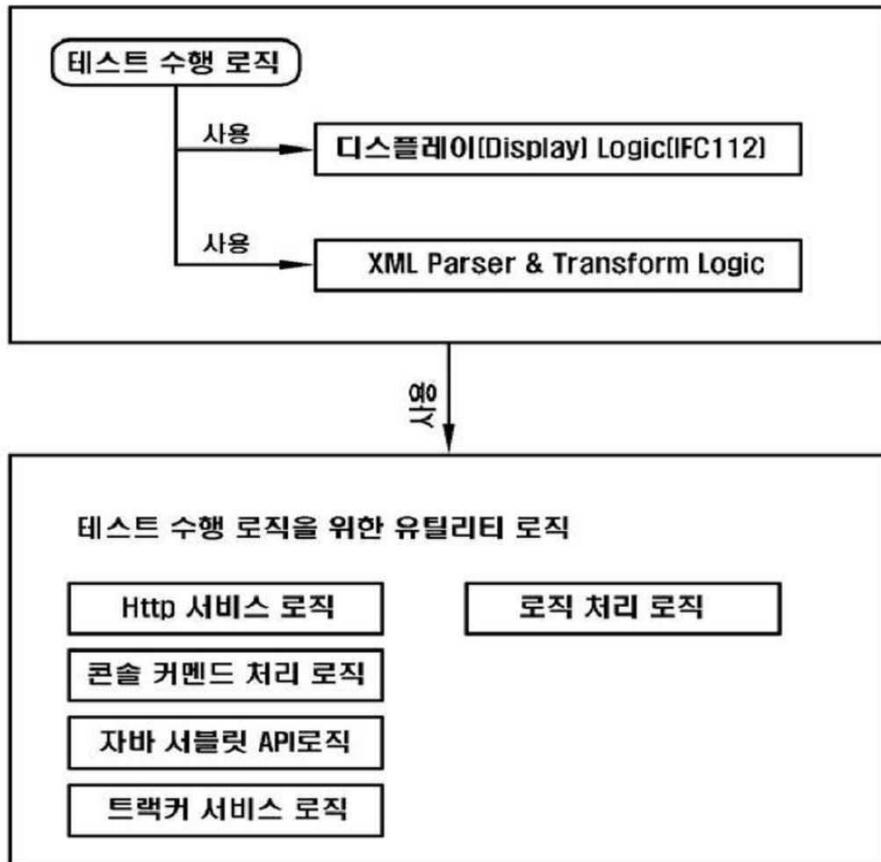
도면9



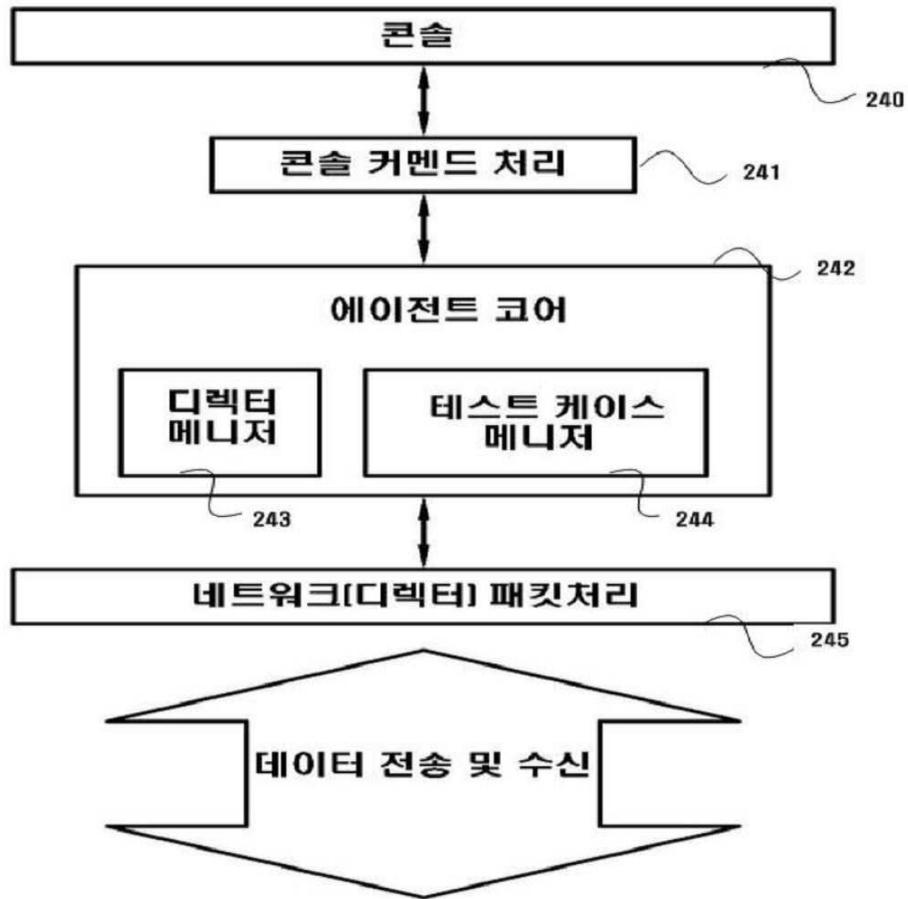
도면10



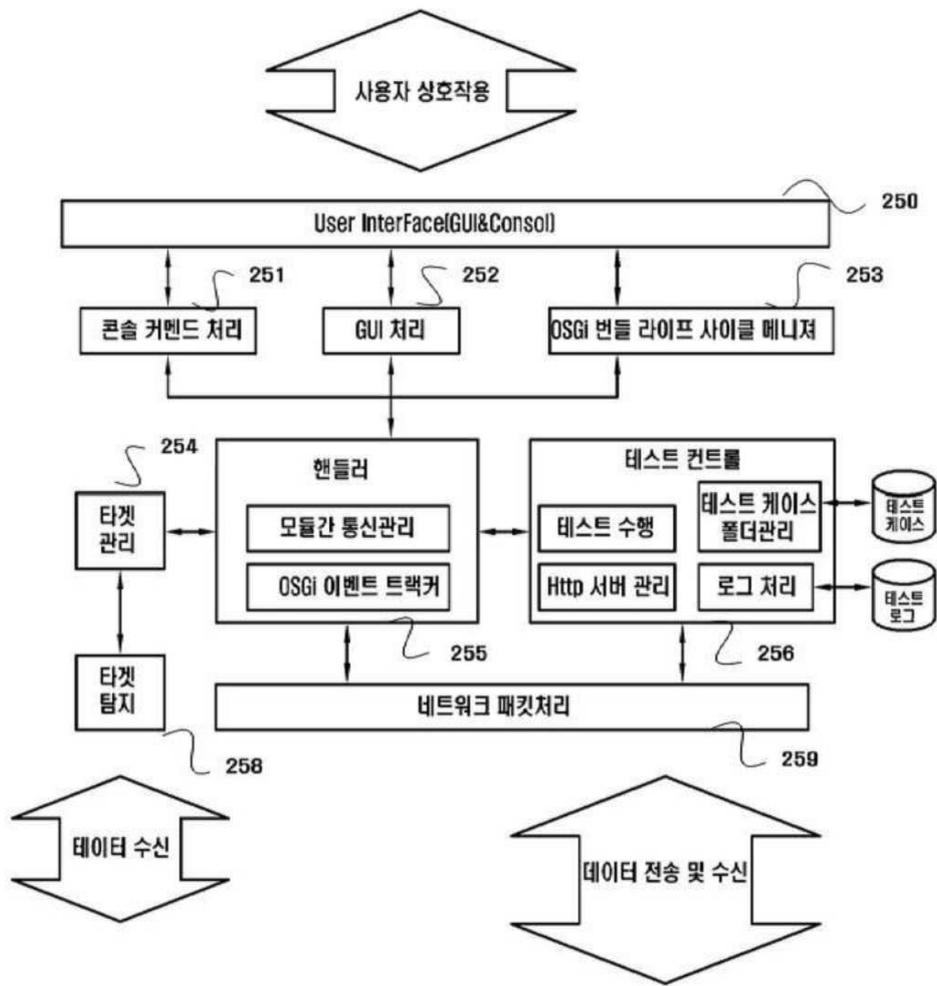
도면11



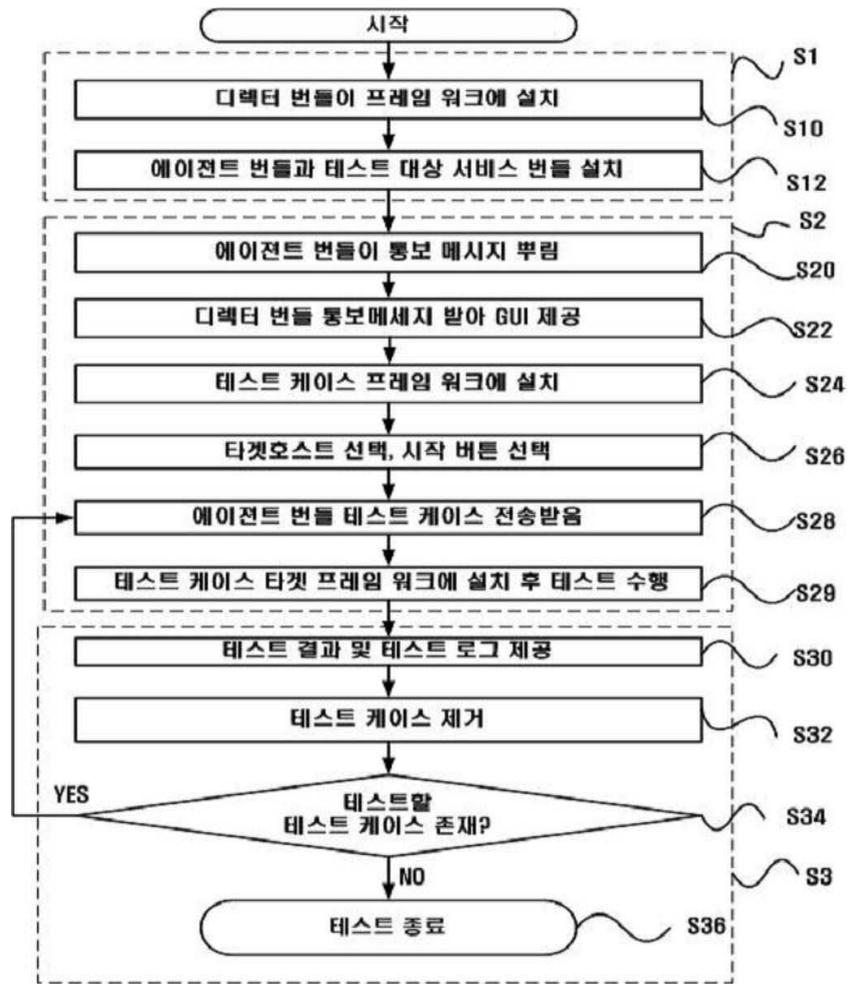
도면12



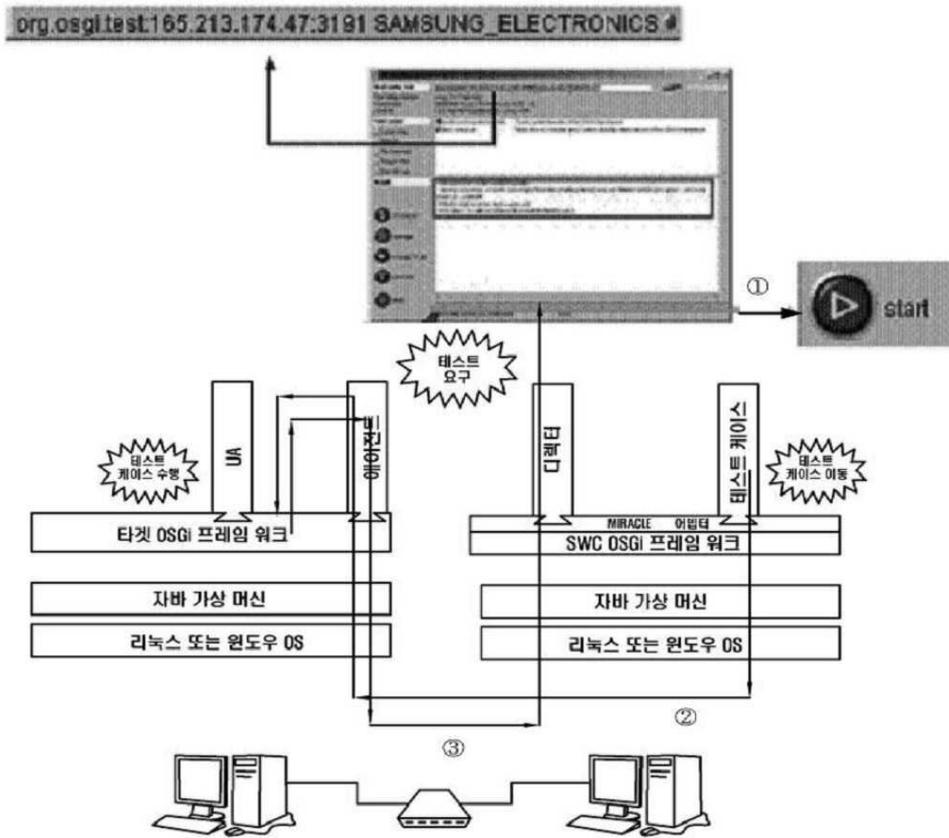
도면13



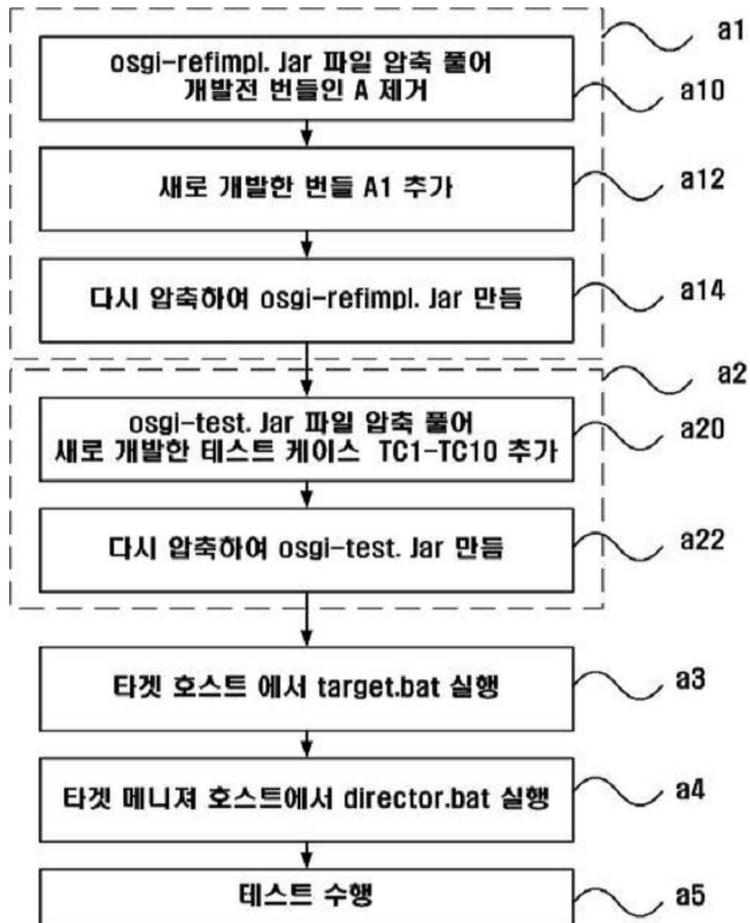
도면14



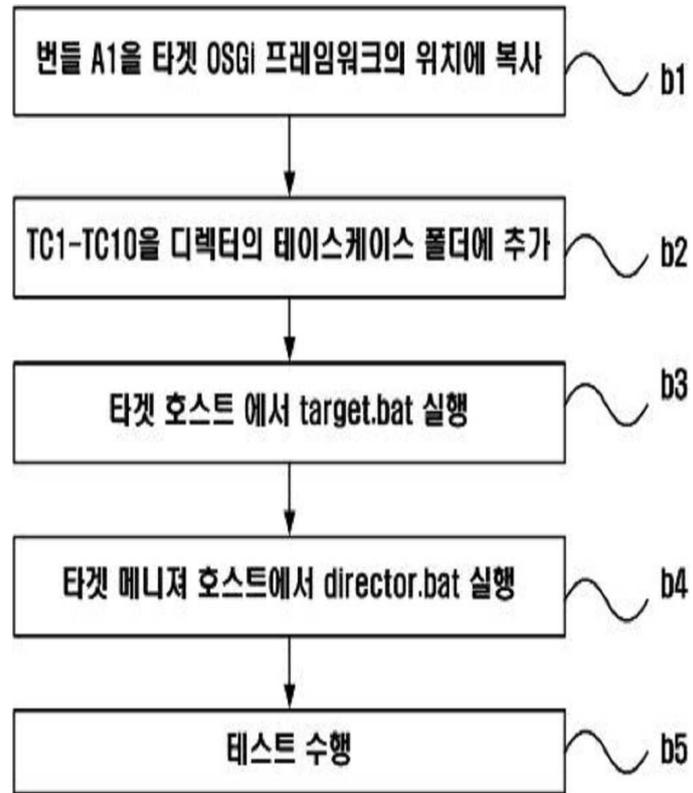
도면17



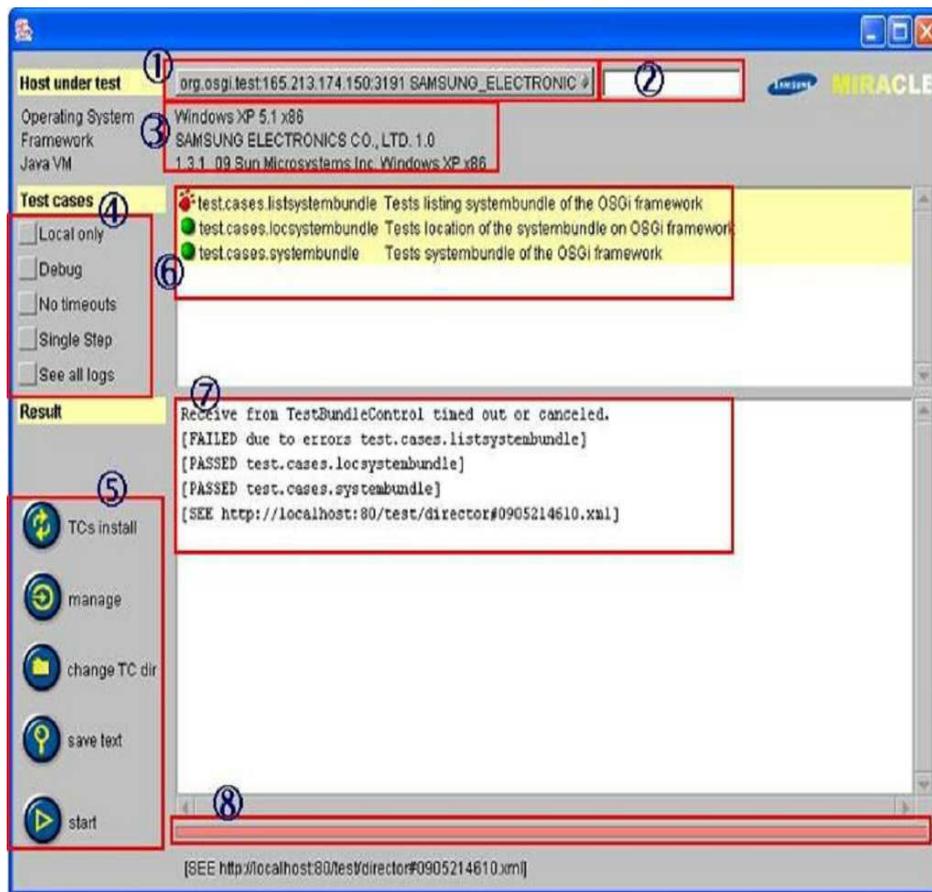
도면18a



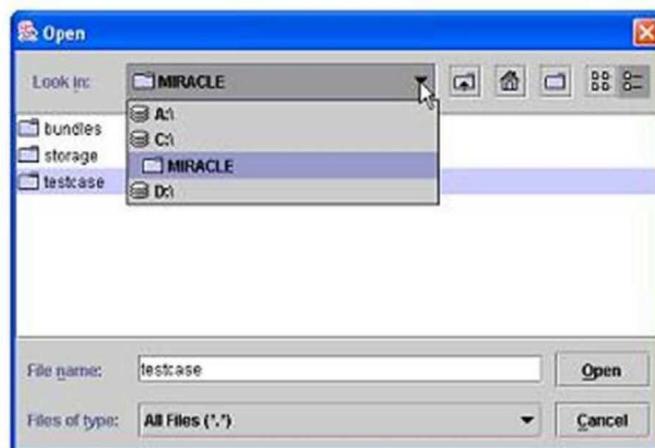
도면18b



도면19



도면20



도면21

