

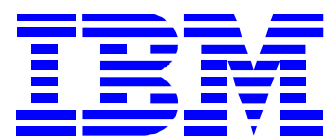
WebSphere Application Server v8.5.5

Liberty 서버 통합관리를 위한 Collective Controller 설정

(2013. 11.)

IBM SWG WebSphere Technical Sales

이정운 과장(juwlee@kr.ibm.com)



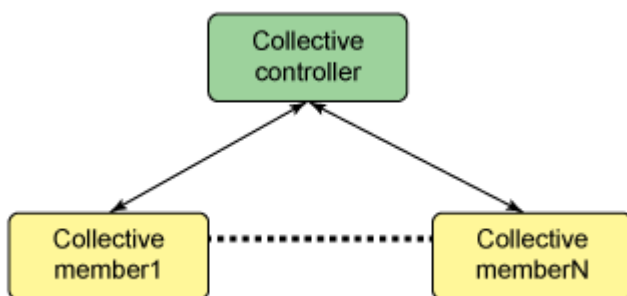
0) Liberty 서버 통합관리를 위한 Collective Controller 설정 간략 소개

안녕하세요 freeman 입니다.

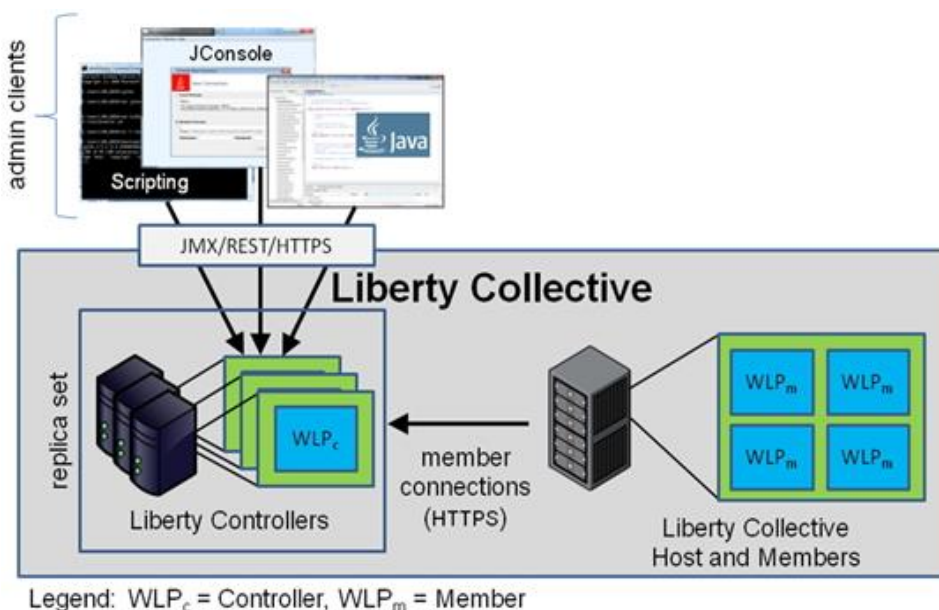
이번 강좌에서 언급드릴 내용은 IBM WAS Liberty 서버 통합관리를 위한 Collective Controller 설정입니다. 이미 지난 강좌에서 언급 드린 것처럼 IBM WAS v8.5.5 에서 가장 많이 변화된 부분은 Liberty 서버의 강화입니다. 특히, Collective Controller 가 생기게 되면서 단순히 1, 2대의 IBM WAS Liberty 서버가 아니라 대용량 환경 또는 클라우드 환경에서 다수의 IBM WAS Liberty 서버를 쉽게 통합 관리할 수 있는 방안을 제공할 수 있습니다.

(단, Collective Controller 를 사용하기 위해서는 Collective Controller 가 설치된 서버에 대한 IBM WAS ND 라이선스가 필요합니다.)

Collective controller 는 그 자체로 IBM WAS Liberty 서버로서 Collective member 로 정의되는 다수의 IBM WAS Liberty 서버를 통합 관리할 수 있습니다.



통합 관리란 말 그대로 등록된 전체 IBM WAS Liberty 서버에 대한 시작/중지와 같은 제어부터, 설정 관리, 고가용성 관리 등이 모두 가능합니다.



Collective controller 에 대한 내용을 보다 심도 깊게 보시고 싶은 분은 하단의 DeveloperWorks article 에 좋은 내용이 쉽게 잘 설명되어 있으니 해당 글을 참고하시기 바라겠습니다.

Introducing the Liberty collective: A cloud-enabled application server management foundation

http://www.ibm.com/developerworks/websphere/techjournal/1309_vignola/1309_vignola.html

Building a large scale WebSphere Application Server Liberty collective topology

http://www.ibm.com/developerworks/websphere/library/techarticles/1309_yu/1309_yu.html

Liberty profile: Collective architecture

http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.nd.doc/ae/cwlp_collective_arch.html

그럼 좀 더 자세한 사항은 이제부터 진행되는 강좌를 참고해주시기 바라면서 지금부터 강좌를 진행하도록 하겠습니다.

1) IBM WAS ND 용 Liberty 서버 설치

0. 이미 언급드린 것처럼 Collective Controller 를 설치하려면 IBM WAS ND 용 Liberty 서버가 필요합니다. 해당 Liberty 서버는 이전 강좌와 같이 압축 해제를 통한 설치도 가능하지만 통합 관리 측면에서 IBM Installation Manager 를 통한 설치도 가능합니다. 지난 강좌에서 이미 압축 해제를 통한 설치를 보여드렸으니 이번 강좌에서는 IM 을 통한 설치로 진행을 하도록 하겠습니다.

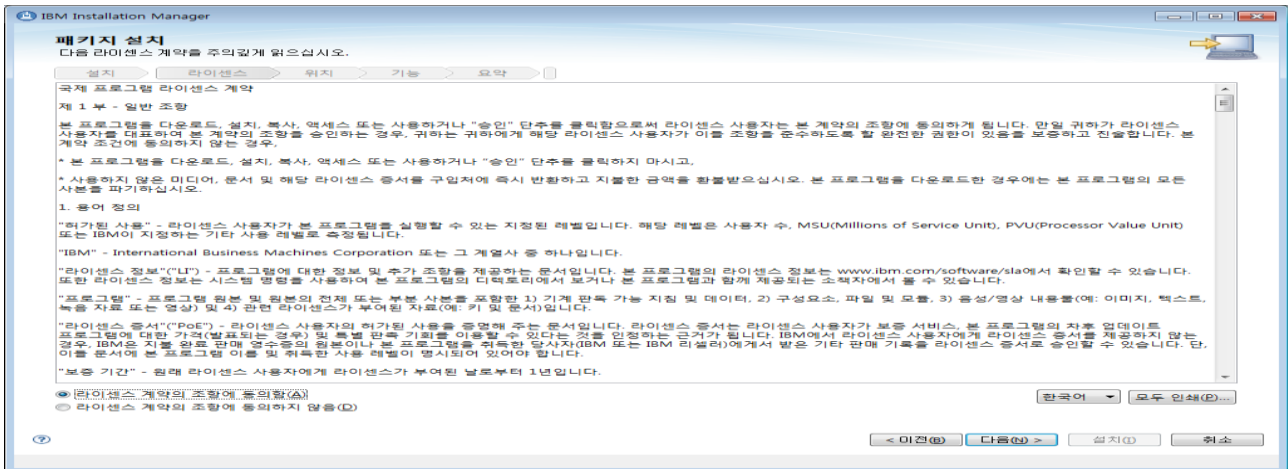
1. IM 으로 설치하기 위하여 IBM Installation Manager 를 구동하고 IBM WAS 용 Liberty 서버를 설치할 이미지 파일이 위치한 저장소를 추가한 후 설치 버튼을 클릭합니다.



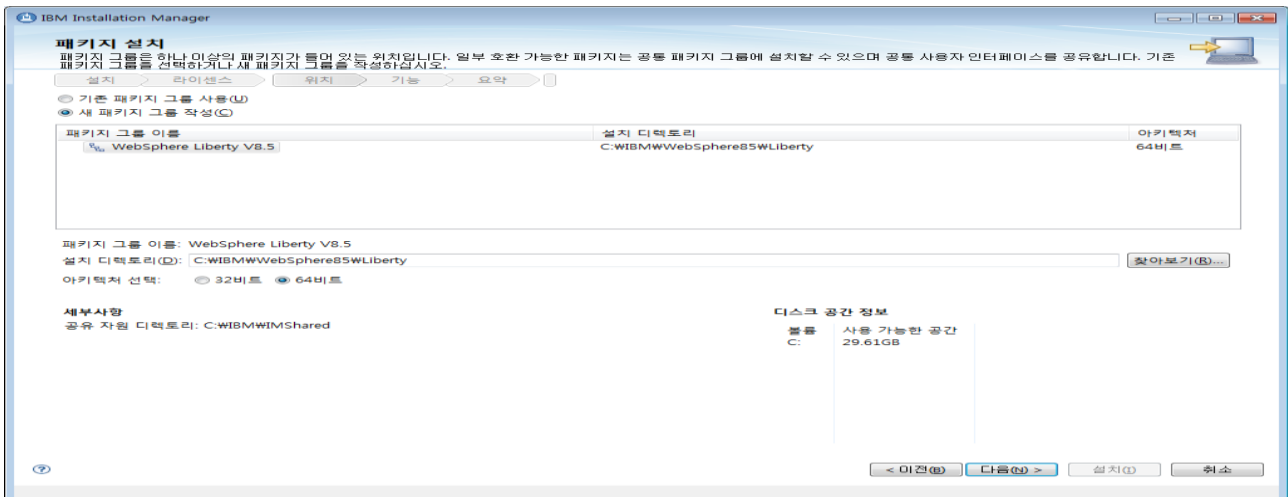
2. 패키지 설치 화면이 나오면 이미 언급한 것처럼 IBM WAS ND 용 Liberty 를 선택합니다.



3. 라이선스 계약을 확인하고 다음을 클릭합니다.



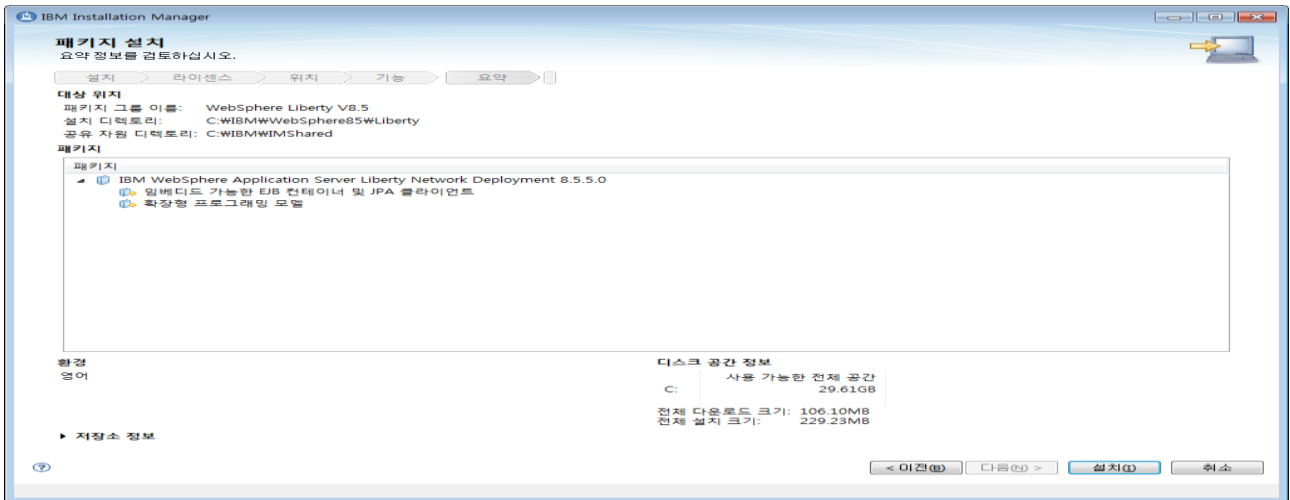
4. 패키지 설치를 위한 디렉토리를 확인하고 다음을 클릭합니다.



5. 설치할 기능을 확인하고 다음을 클릭합니다.



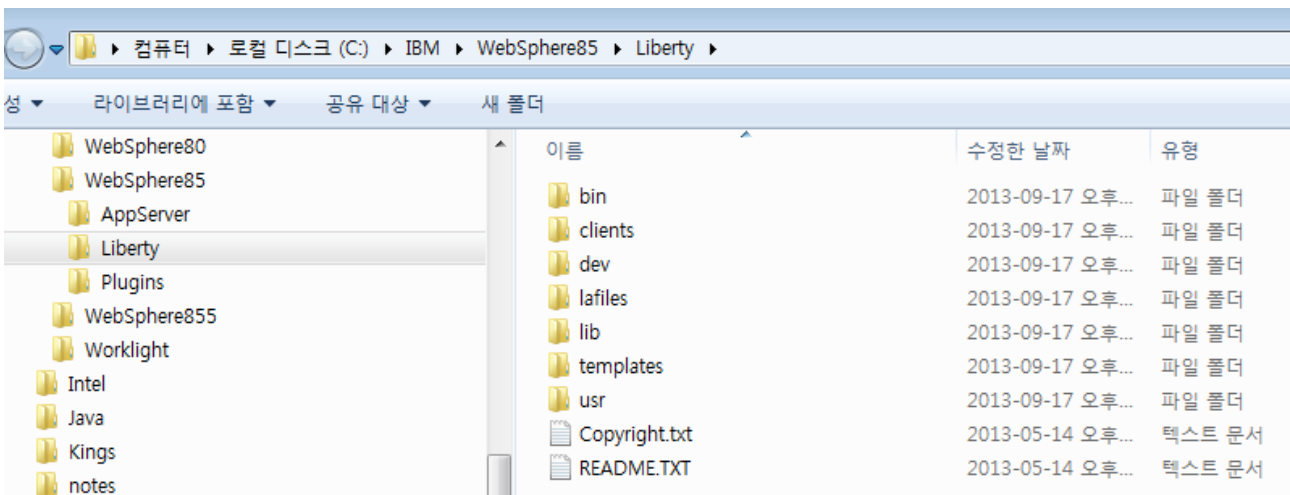
6. 마지막으로 요약 정보를 클릭하고 설치를 누르면 IBM WAS ND 용 Liberty 서버가 설치됩니다.



7. 패키지 설치가 완료되면 하단과 같은 메시지를 확인할 수 있으며 완료를 클릭합니다.



8. 성공적으로 IBM WAS ND 용 Liberty 서버가 설치되면 하단과 같은 디렉토리가 생성된 것을 확인할 수 있습니다. (보시면 아시겠지만 jar 압축 해제 방식으로 설치한 것과 동일한 구조입니다.)



2) Liberty 서버 통합관리를 위한 Collective Controller 설정

1. Liberty 서버 통합관리를 위한 Collective Controller 설정을 위하여 Collective Controller 로 사용할 IBM WAS Liberty 서버를 하나 생성합니다. IBM WAS Liberty 서버의 생성은 Liberty 설치 폴더\bin 디렉토리에서 "server create 서버명" 과 같은 단순 명령으로 생성가능 합니다.

```
D:\Weclipses\wlp\bin>server create collective01
collective01 서버가 작성되었습니다.
```

2. Collective Controller 로 사용할 IBM WAS Liberty 서버 생성이 완료되면 해당 서버를 대상으로 Collective Controller 설정을 생성합니다.

(collective create collective01 -keystorePassword=123456)

```
D:\Weclipses\wlp\bin>collective create collective01 --keystorePassword=123456
통합을 설정하는 데 필요한 인증서를 작성하는 중...
시간이 조금 걸릴 수 있습니다.
제어기 루트 인증서가 생성되었습니다.
멤버 루트 인증서가 생성되었습니다.
서버 ID 인증서가 생성되었습니다.
HTTPS 인증서가 생성되었습니다.
```

collective01의 통합 제어기 구성이 설정되었습니다.

다음 행을 server.xml에 추가하여 사용 가능하게 설정하십시오.

```
<featureManager>
  <feature>collectiveController-1.0</feature>
</featureManager>

<!-- Define the host name for use by the collective.
     If the host name needs to be changed, the server should be
     removed from the collective and re-joined or re-replicated. -->
<variable name="defaultHostName" value="kr050578" />

<!-- TODO: Set the security configuration for Administrative access -->
<quickStartSecurity userName="" userPassword="" />

<!-- clientAuthenticationSupported set to enable bidirectional trust -->
<ssl id="defaultSSLConfig"
     keyStoreRef="defaultKeyStore"
     trustStoreRef="defaultTrustStore"
     clientAuthenticationSupported="true" />

<!-- inbound <HTTPS> keystore -->
<keyStore id="defaultKeyStore" password="{xor}bm1sa2pp"
          location="{server.config.dir}/resources/security/key.jks" />

<!-- inbound <HTTPS> truststore -->
<keyStore id="defaultTrustStore" password="{xor}bm1sa2pp"
          location="{server.config.dir}/resources/security/trust.jks" />

<!-- server identity keystore -->
<keyStore id="serverIdentity" password="{xor}bm1sa2pp"
```

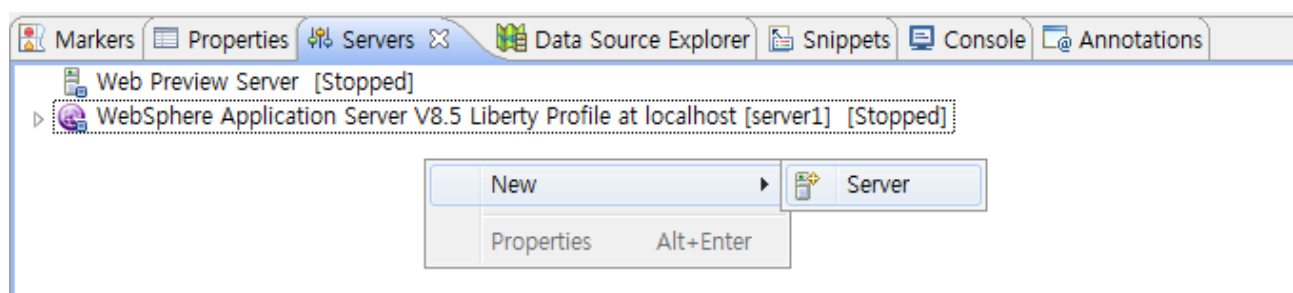
3. Collective Controller 설정을 생성되면 지시된 데로 해당 설정을 server.xml 에 추가합니다.

4. 또한, quickStartSecurity 선언부분에 공백으로 되어 있는 userName 와 userPassword 를 추가합니다.

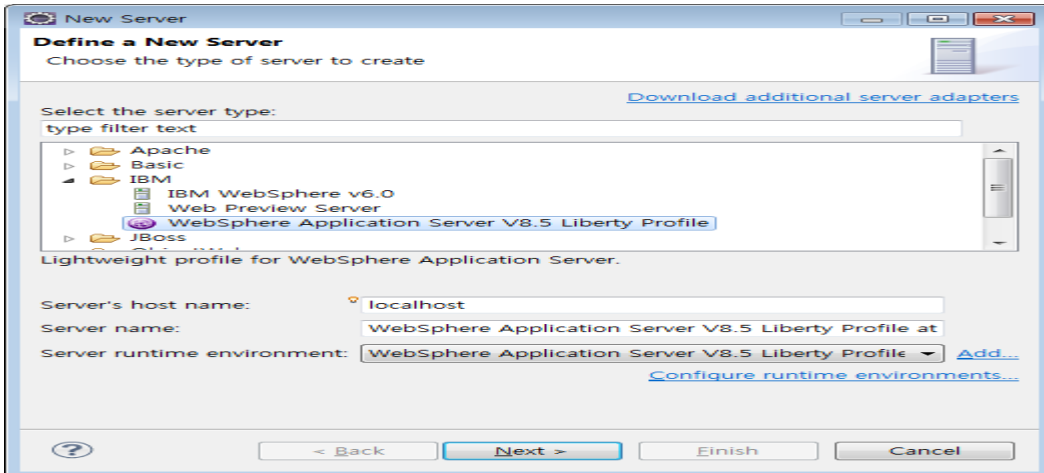
```
<server description="new server">↵
↵
  <!-- Enable features -->↵
  <featureManager>↵
    <feature>jsp-2.2</feature>↵
    <feature>collectiveController-1.0</feature>↵
  </featureManager>↵
↵
  <httpEndpoint id="defaultHttpEndpoint"↵
    host="localhost"↵
    httpPort="9080"↵
    httpsPort="9443" />↵
↵
  <!-- Define the host name for use by the collective.↵
    If the host name needs to be changed, the server should be↵
    removed from the collective and re-joined or re-replicated. -->↵
  <variable name="defaultHostName" value="kr050578" />↵
↵
  <!-- TODO: Set the security configuration for Administrative access -->↵
  <quickStartSecurity userName="wasadm" userPassword="wasadm" />↵
↵
  <!-- clientAuthenticationSupported set to enable bidirectional trust -->↵
  <ssl id="defaultSSLConfig"↵
    keyStoreRef="defaultKeyStore"↵
    trustStoreRef="defaultTrustStore"↵
    clientAuthenticationSupported="true" />↵
↵
  <!-- inbound (HTTPS) keystore -->↵
  <keyStore id="defaultKeyStore" password="{xor}bm1sa2pp"↵
    location="${server.config.dir}/resources/security/key.jks" />↵
```

4. Collective Controller 로 사용하기 위해서 생성된 IBM WAS Liberty 서버를 좀 더 편리하게 관리하기 위해 eclipse 와 연동작업을 수행합니다.

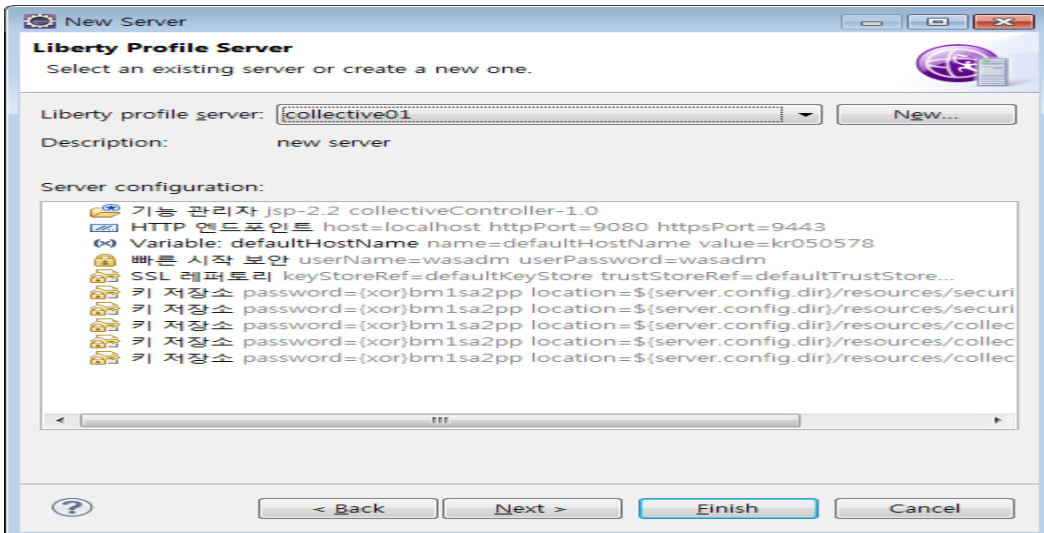
5. eclipse 에 IBM WAS Liberty 서버를 연동하기 위해서는 Servers 탭에서 마우스 우 클릭하여 New > Server 를 클릭합니다.



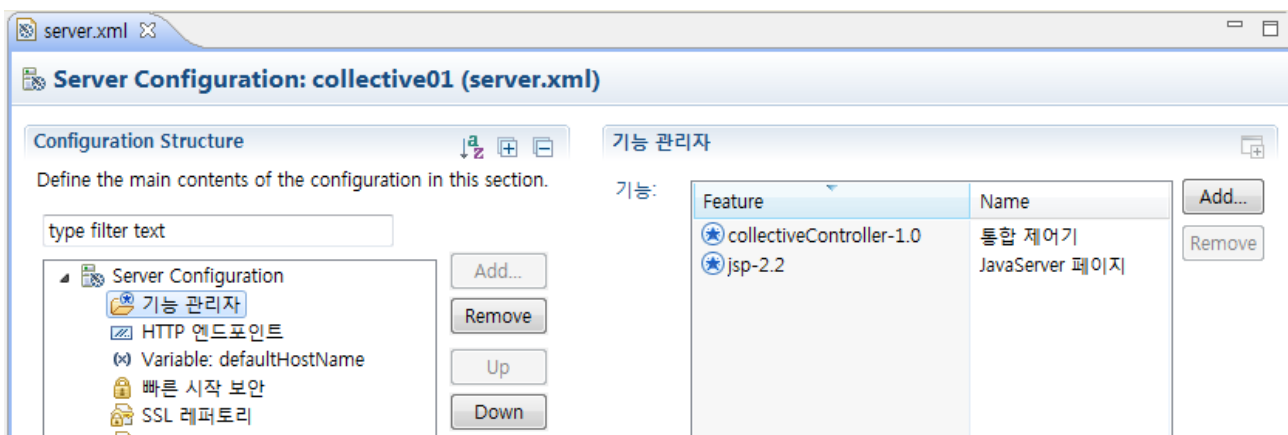
6. 서버 타입을 선택하는 화면이 나오면 IBM WAS Liberty profile 을 선택하고 Next 를 클릭합니다.



7. Collective Controller 용으로 만들어둔 collective01 서버를 선택하고 Finish 를 클릭하여 완료합니다.



8. 이와 같이 하면 Collective Controller 용으로 만들어둔 collective01 서버를 eclipse 를 통해서 손쉽게 관리 가능합니다.



9. Collective Controller 설정이 완료되었으므로 이제 Collective Member 로 등록해서 사용할 IBM WAS Liberty 서버를 생성합니다. (실제 서비스를 수행할 Liberty 서버)

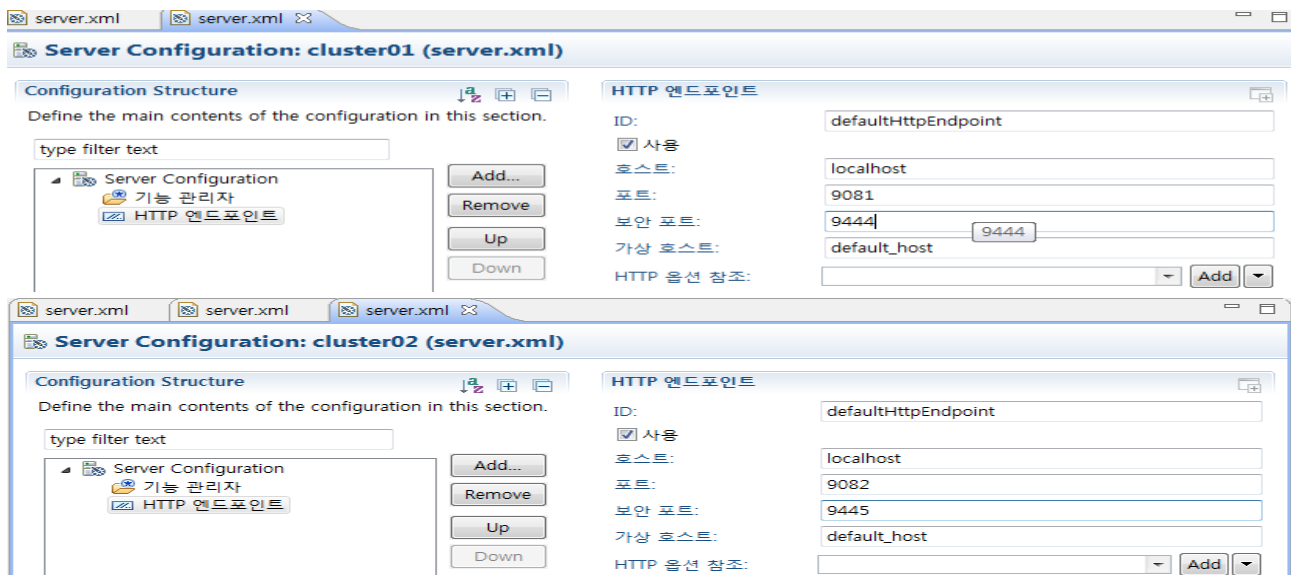
10. Collective Member 로 등록해서 사용할 IBM WAS Liberty 서버를 두 개 정도 생성합니다.
(단일 IBM WAS Liberty 서버가 아니라 클러스터 구성을 해서 사용하기 위해서 2개 정도 생성한 것 입니다.)

```
D:\weclipses\wlp\bin>server create cluster01
cluster01 서버가 작성되었습니다.

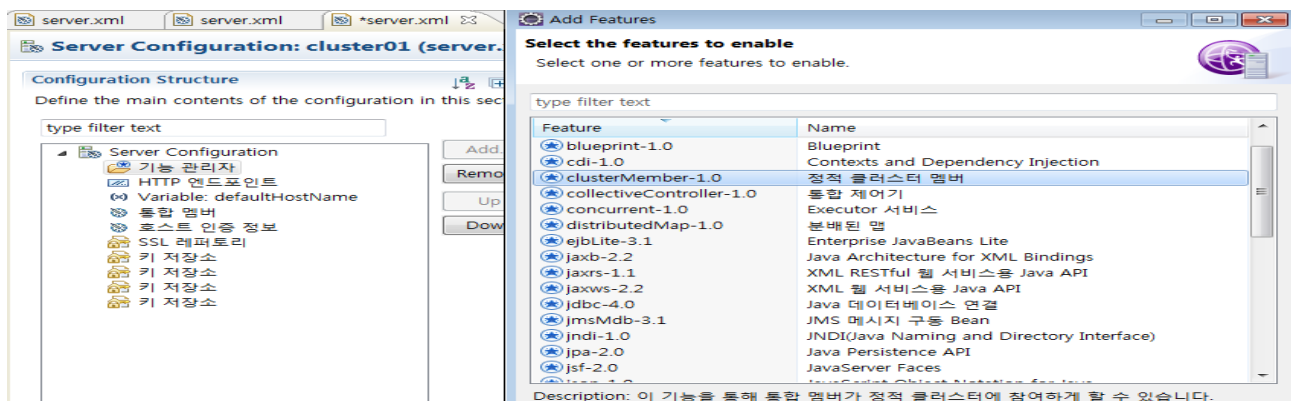
D:\weclipses\wlp\bin>server create cluster02
cluster02 서버가 작성되었습니다.
```

11. 이전 Collective Member 와 동일하게 작업의 편의성을 위해 eclipse 와 연동작업을 수행합니다.

12. 실제로 서비스할 때 서비스 포트가 충돌 되지 않게 서비스 포트를 변경해 줍니다.



13. 또한, 두 대의 IBM WAS Liberty 서버 모두 클러스터 멤버로 등록되도록 server.xml 의 기능 관리자에 서 clusterMember-1.0 feature 를 추가합니다.



14. 생성된 Liberty 서버를 Collective Controller 와 연결하기 위하여 하단과 같은 명령을 이용해서 join 작업을 수행합니다. 이때 Collective Controller 는 반드시 start 되어 있어야 합니다.

```
(collective join cluster01 -host=localhost -port=9443 -user=wasadm -password=wasdm -keystorePassword=123456
```

여기서 port 는 Collective Controller 의 HTTPS 를 의미하며 user 와 password 는 Collective Controller 의 quickStartSecurity 에 선언한 user 와 password 를 의미합니다.)

```
C:\IBM\WebSphere85\Liberty\bin>collective join cluster01 --host=localhost --port=9443 --user=wasadm --password=wasadm --keystorePassword=123456
대상 제어기 localhost:9443과<와> 통합을 결함하는 중...
시간이 조금 걸릴 수 있습니다.

대상 서버와의 SSL 신뢰가 설정되지 않았습니다.

인증 체인 정보:
인증 [0]
주제 DN: CN=kr050578, OU=collective01, O=ibm, C=us
발행자 DN: OU=controllerRoot, O=00d6bc7b-0f28-4c5a-8a5f-26dd34672185, DC=com.ibm
.ws.collective
일련 번호: 251.647.565.770.858
만기: 18. 10. 13 오전 11:09
SHA-1 요약: EC:D6:D8:B4:5D:97:63:85:7D:33:E1:44:5A:27:58:3E:5B:F5:E8:7F
MD5 요약: 7C:63:E9:BF:50:72:67:23:70:74:91:A7:58:ED:1A:40

인증 [1]
주제 DN: OU=controllerRoot, O=00d6bc7b-0f28-4c5a-8a5f-26dd34672185, DC=com.ibm
.ws.collective
발행자 DN: OU=controllerRoot, O=00d6bc7b-0f28-4c5a-8a5f-26dd34672185, DC=com.ibm
.ws.collective
일련 번호: 251.643.695.560.432
만기: 38. 10. 8 오전 11:09
SHA-1 요약: C1:A6:BD:8D:F5:EA:29:BB:87:35:A7:9D:F6:52:A5:44:EF:00:17:55
MD5 요약: 6B:ED:E3:57:3C:8A:8C:E7:59:16:31:16:72:DC:82:0C

위의 인증 체인을 승인하시겠습니까? <예/아니 오>
제어기에 대한 MBean 요청이 완료되었습니다.

cluster01 서버의 통합이 결함되었습니다.
```

15. Collective Controller 설정 생성과 마찬가지로 지시대로 join 후에 나온 내용을 각 Liberty 서버의 server.xml 에 추가합니다.

```
server.xml  server.xml  *server.xml  server.xml

<server description="new server">
  <!-- Enable features -->
  <featureManager>
    <feature>jsp-2.2</feature>
    <feature>collectiveMember-1.0</feature>
    <feature>clusterMember-1.0</feature>
  </featureManager>

  <httpEndpoint id="defaultHttpEndpoint"
    host="localhost"
    httpPort="9081"
    httpsPort="9444" />

  <!-- Define the host name for use by the collective.
    If the host name needs to be changed, the server should be
    removed from the collective and re-joined or re-replicated. -->
  <variable name="defaultHostName" value="kr050578" />

  <!-- Connection to the collective controller -->
  <collectiveMember controllerHost="localhost"
    controllerPort="9443" />

  <hostAuthInfo rpcHost="kr050578" rpcUser="kr050578" rpcUserPassword="{xor}bDo7PCk5LWs=" />

  <!-- clientAuthenticationSupported set to enable bidirectional trust -->
  <ssl id="defaultSSLConfig"
    keyStoreRef="defaultKeyStore"
    trustStoreRef="defaultTrustStore"
    clientAuthenticationSupported="true" />
```

16. 추가로 타켓 서버에 대한 제어 작업을 수행할 수 있게 하단의 구문을 Member 로 등록될 Liberty 서버의 server.xml 에 추가합니다. 여기서 rpcUser 와 rpcUserPassword 는 windows 의 경우 admin 권한 이 있는 windows 사용자 정보입니다.

```
<hostAuthInfo rpcHost="hostname" rpcUser="username" rpcUserPassword="password" />
```

Or

Member 를 Collective Controller 에 join 할 때 하단과 같이 명령을 수행하면 해당 구문이 자동으로 생성됩니다.

```
collective join member01 --host=localhost --port=9443 --user=wasadm --password=wasadm --keystorePassword=123456 --rpcUser=username --rpcUserPassword= password --hostName=localhost)
```

참고: Starting and stopping a Liberty collective member

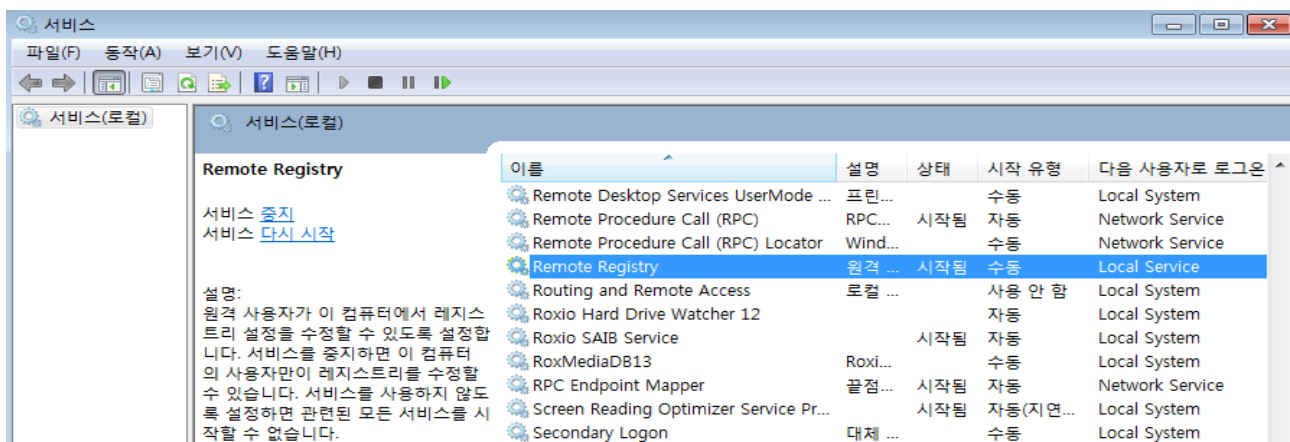
http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.doc/ae/ragt_wlp_servercomand.html?resultof=%22%68%6f%73%74%41%75%74%68%49%6e%66%6f%22%20%22%68%6f%73%74%61%75%74%68%69%6e%66%6f%22%20

17. 타켓 서버를 대상으로 remote operaion 을 수행할 경우 RXA(Tivoli Remote Execution and Access) 를 내부적으로 사용하는데 이 때 보안 관련 부분에 대한 해제가 필요합니다. 이 부분이 반드시 체크되어야 합니다.

참고: Setting up RXA for Liberty collective operations

http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.doc/ae/twlp_set_rxa.html

테스트 해본 결과 Windows 에서는 UAC disable 과 하단과 같이 Remote Registry 서비스만 시작되어 있어도 RXA 사용에는 문제가 없는 것으로 보입니다.



18. 이와 같이 구성하면 Liberty 서버 통합관리를 위한 Collective Controller 설정과 Collective Member 의 join 까지 모든 작업이 완료된 것입니다.

3) Collective Controller 를 통한 Liberty 서버 통합관리

1. Collective Controller 를 통한 Liberty 서버 통합관리를 위해서는 JMX 형태로 관리 요청을 해야 합니다. 이를 위해서 IBM WAS 는 jython 을 통한 다양한 script 를 제공할 뿐만 아니라 jconsole 을 통해서 손쉽게 할수 있는 방안도 같이 제공합니다. 이번 강좌에서는 jconsole 을 통해서 여러대의 Member 를 Collective Controller 를 통해 관리하는 모습을 보여드리도록 하겠습니다.

2. Collective Controller 를 통한 관리를 위해 jconsole 을 수행하는데 SSL 로 통신해야 하기 때문에 하단과 같은 형식으로 jconsole 을 수행합니다.

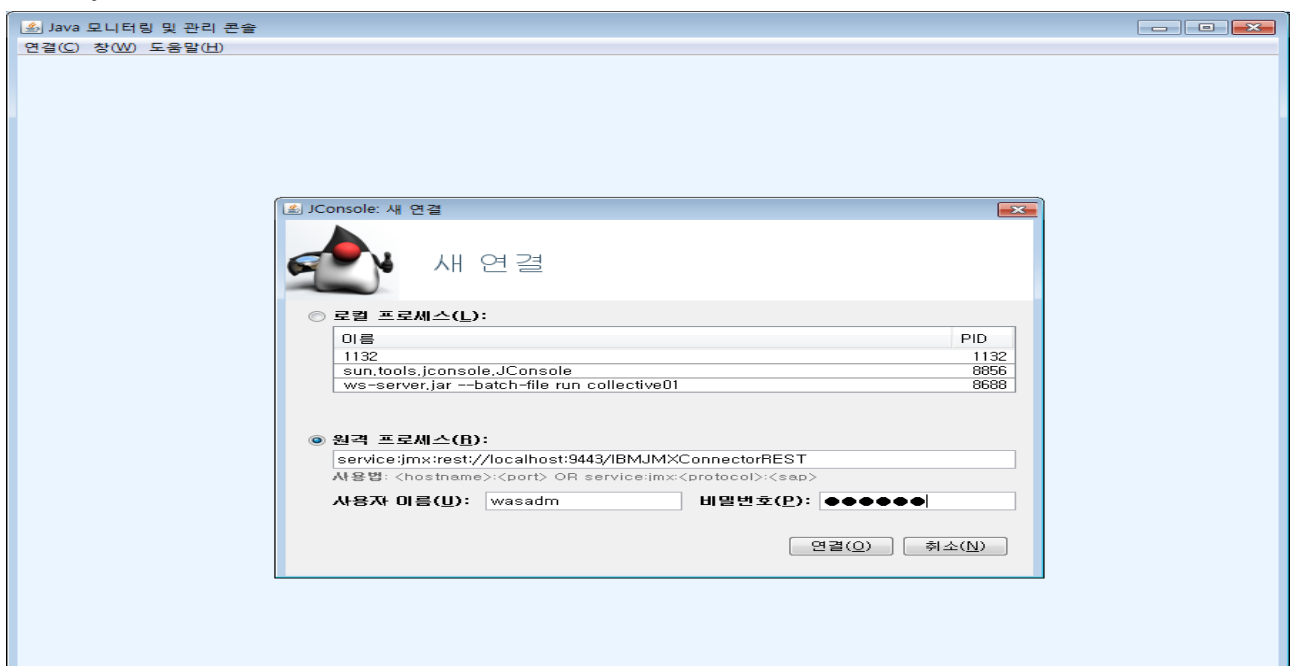
```
(jconsole -J-Djava.class.path=C:\WIBM\WebSphere85\AppServer\java_1.7_64\lib\jconsole.jar;C:\WIBM\WebSphere85\AppServer\java_1.7_64\lib\tools.jar;C:\WIBM\WebSphere85\Liberty\clients\wrestConnector.jar -J-Djavax.net.ssl.trustStore=C:\WIBM\WebSphere85\Liberty\usr\servers\collective01\resources\security\keyjks -J-Djavax.net.ssl.trustStorePassword=123456 -J-Djavax.net.ssl.trustStoreType=jks -J-Dcom.ibm.ws.jmx.connector.client.disableURLHostnameVerification=true
```

여기서 각 path 는 환경에 적합하게 변경되어야 하며 추가적으로 javax.net.ssl.trustStore 는 Collective Controller 생성시에 기본으로 생성되는 keyjks 위치를 명시하면 됩니다.)

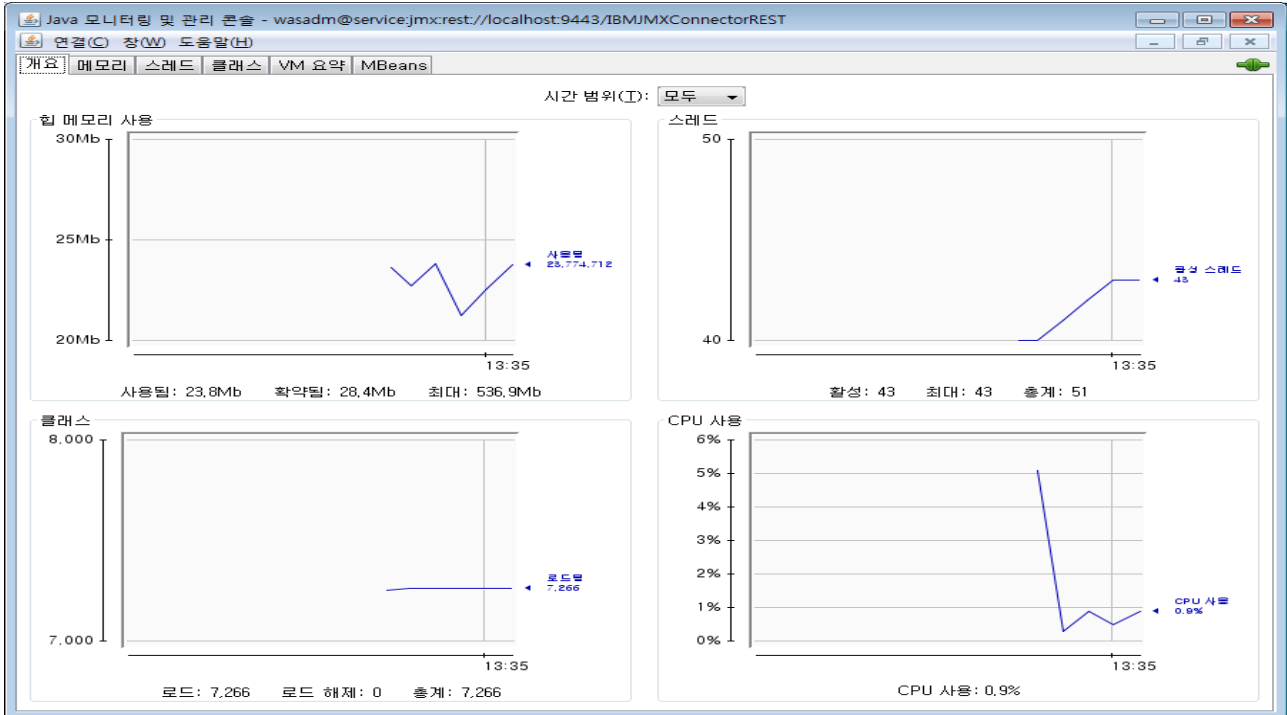
3. Jconsole 이 정상적으로 수행되면 remote 로 접근하기 위하여 접근 주소와 사용자 이름/비밀번호 등을 입력하고 연결 버튼을 클릭합니다. 여기서 사용자 이름은 Collective Controller 에 등록된 사용자 이름을 의미합니다.

(또는, 로컬 프로세스로 Collective Controller 로 접속하여 Insecure 모드로 관리 작업을 수행할 수도 있습니다.)

service:jmx:rest://localhost:9443/IBMJMXConnectorREST

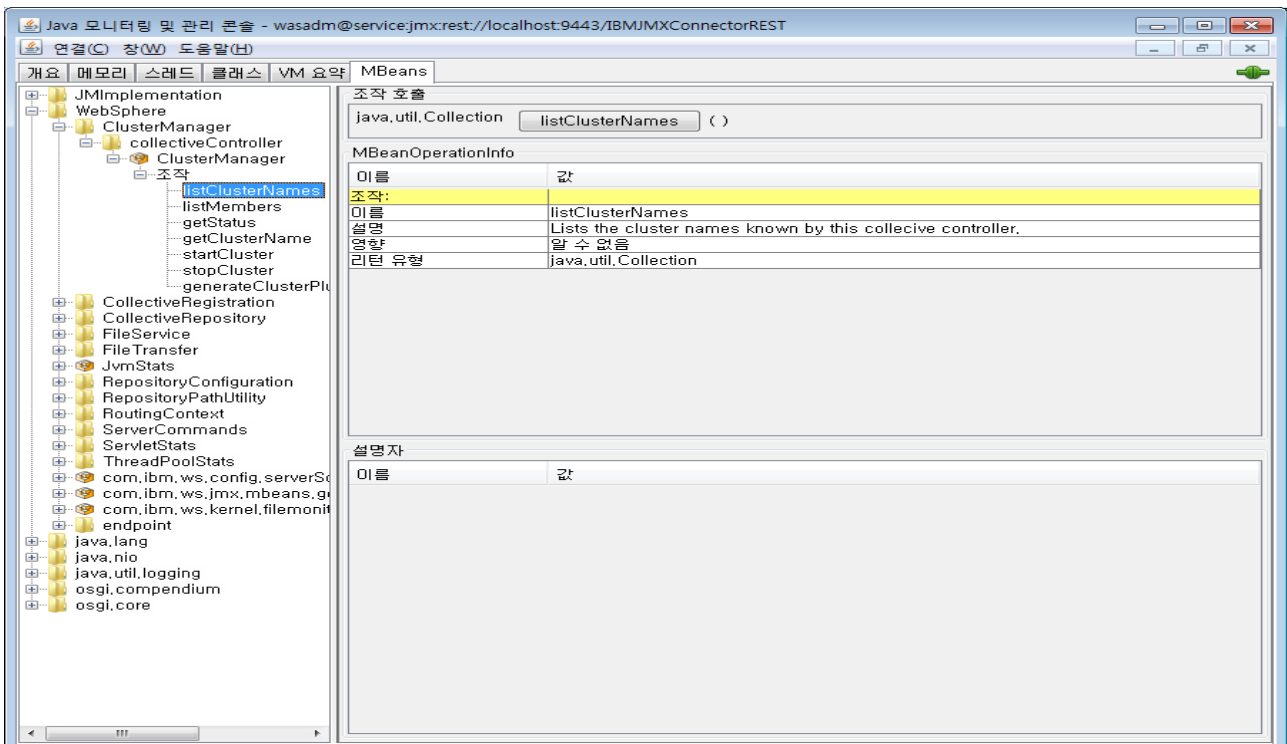


4. 정상적으로 연결이 되었으면 하단과 같은 Jconsole 기본 모니터링 화면을 확인할 수 있습니다.

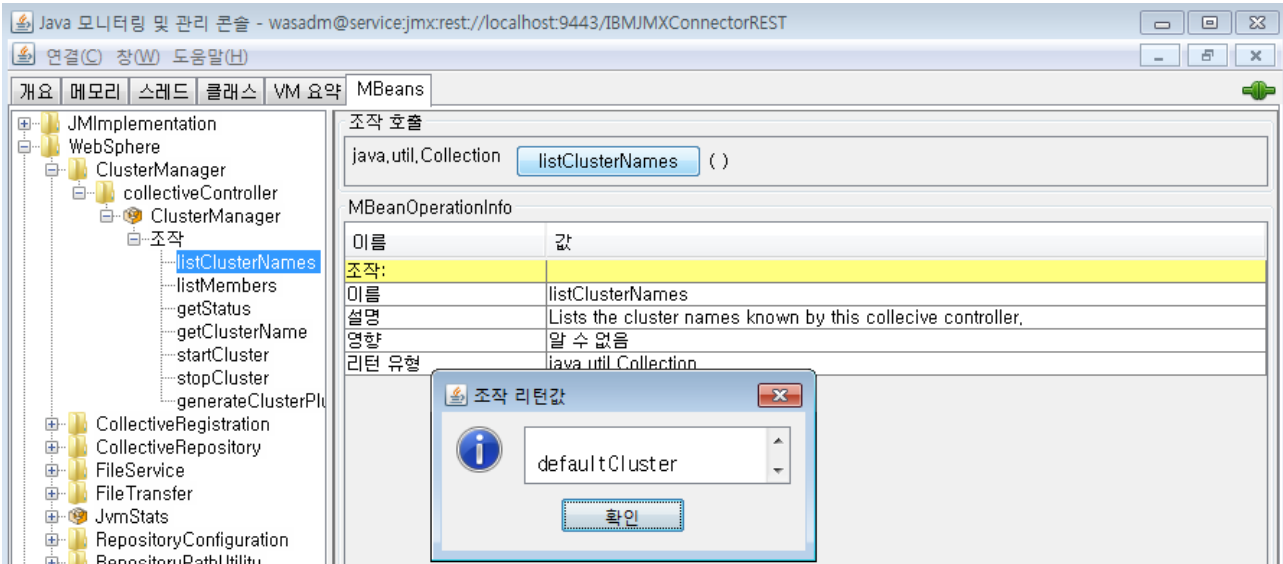


5. 이에 추가하여 통합관리를 목적으로 Jconsole 을 사용하기 위해서는 MBeans 카테코리의 명령들을 사용하면 됩니다.

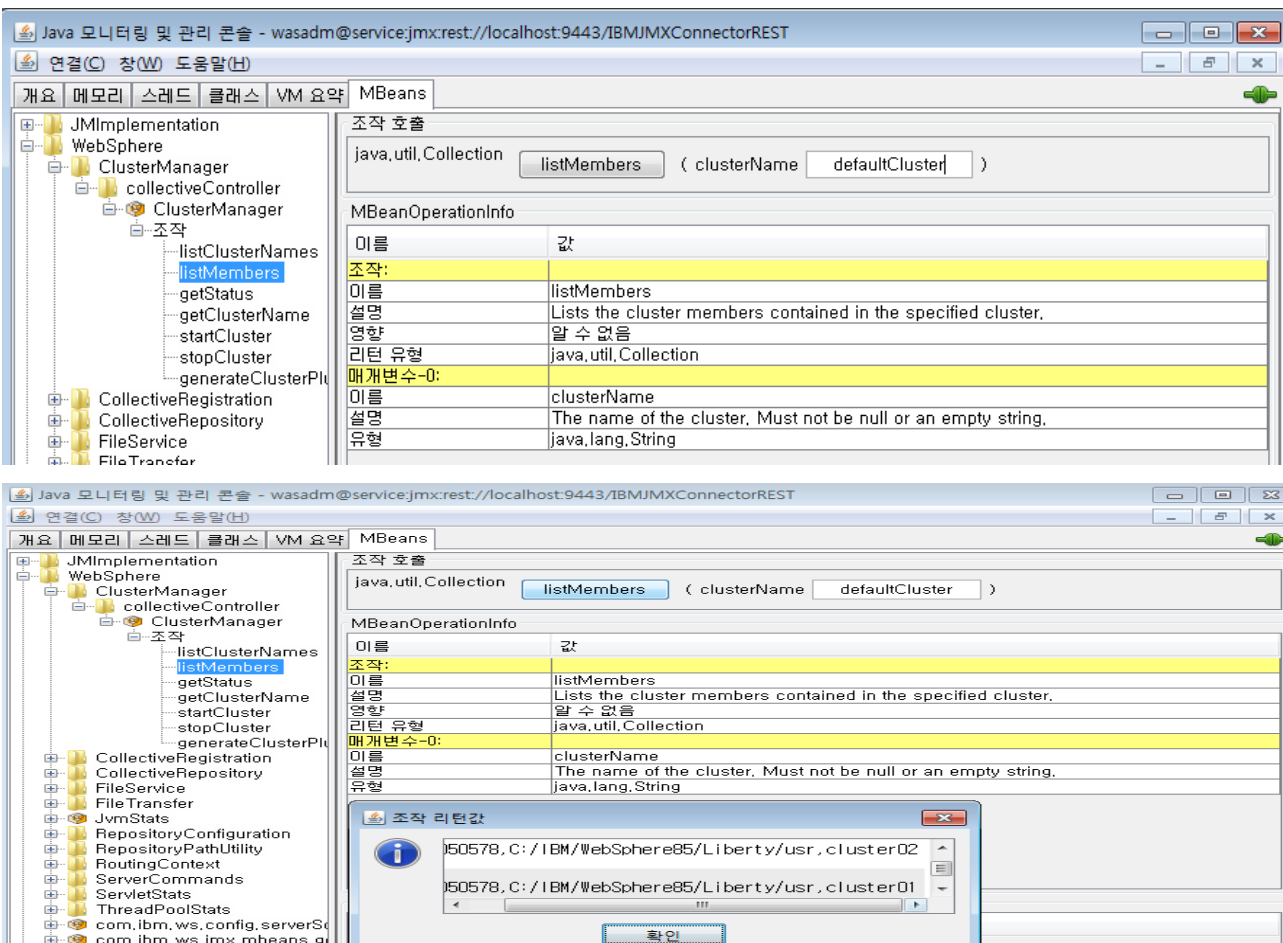
6. MBeans > WebSphere > ClusterManager > collectiveController > ClusterManager 메뉴로 가면 조작할 수 있는 여러 명령들의 리스트를 확인할 수 있습니다.



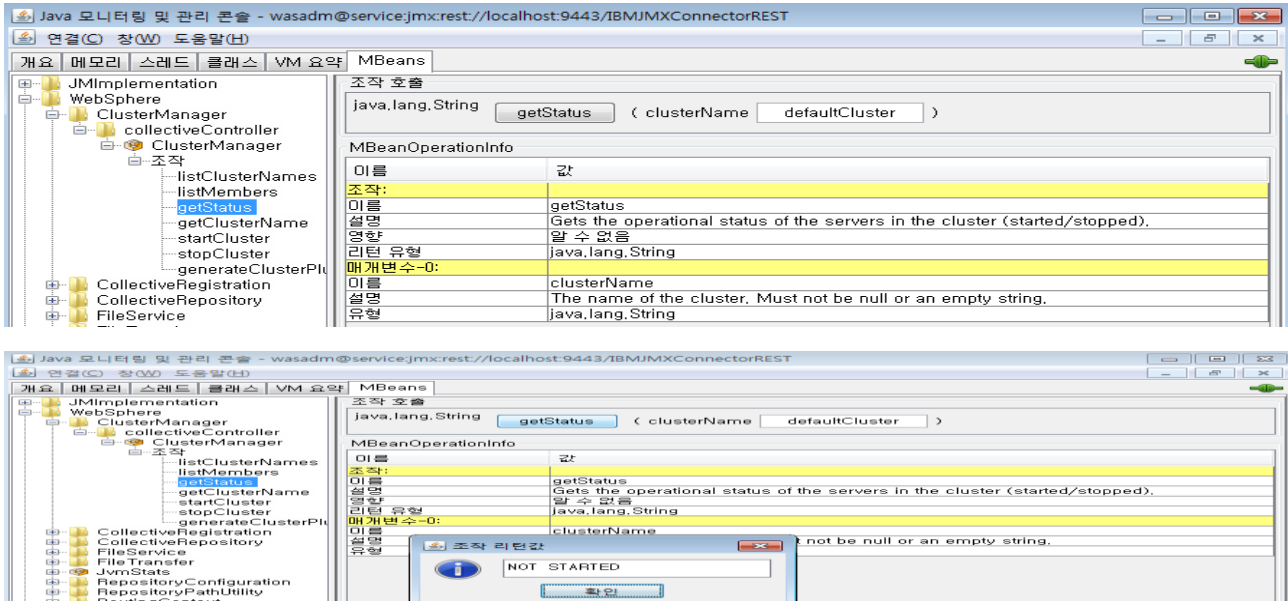
7. 예를 들어 listClusterNames 를 선택하고 버튼을 클릭하면 지금 Collective Controller 에 등록된 클러스터 이름을 확인할 수 있습니다.(Member 를 클러스터로 등록하고 별도로 이름을 주지 않았으므로 defaultCluster 라는 이름을 확인할 수 있습니다.)



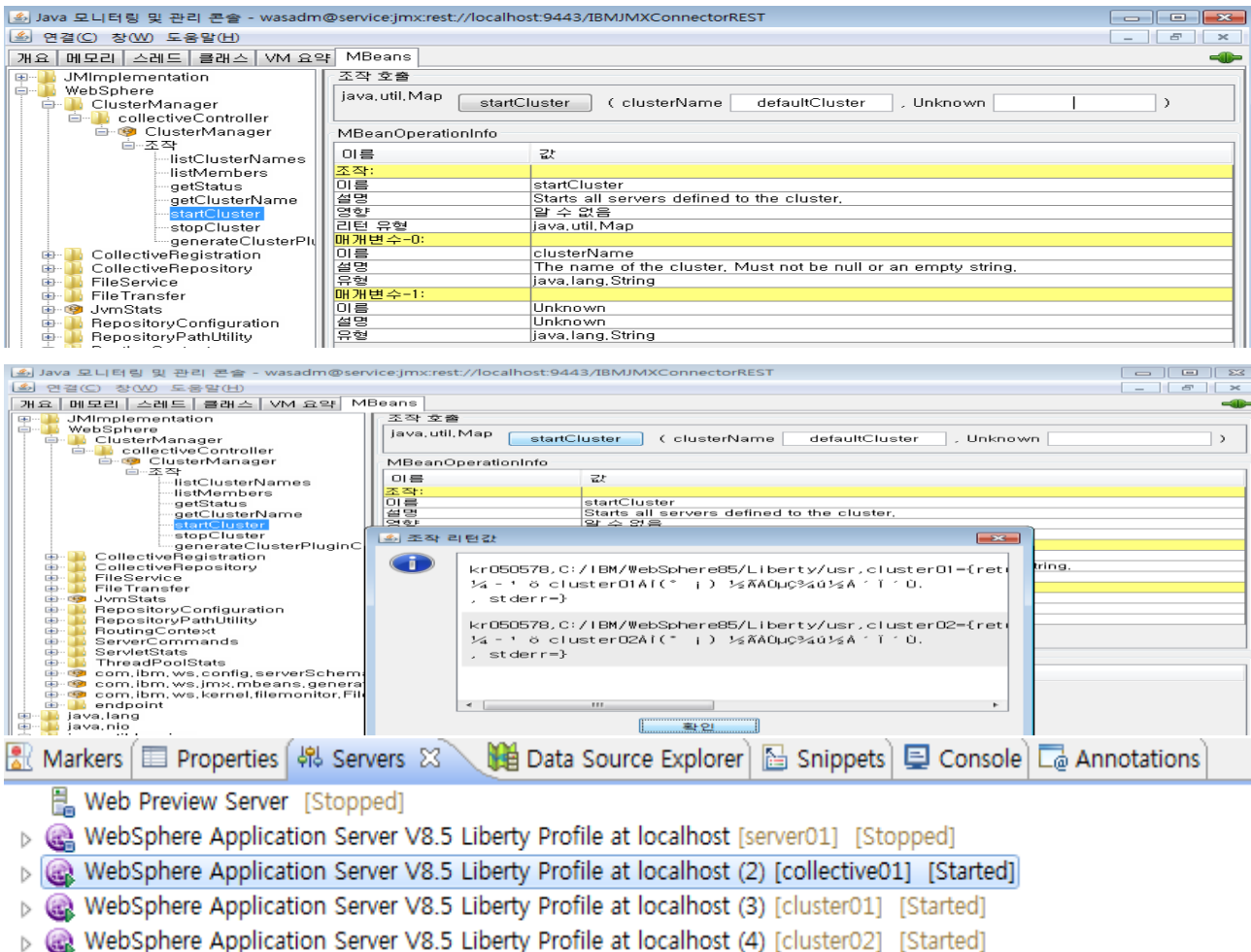
8. 이렇게 확인된 클러스터 이름을 인자로 입력하고 listMembers 명령을 수행하게 되면 해당 클러스터로 등록된 두 개의 Member 서버인 cluster01, cluster02 를 확인할 수 있습니다.



9. 이와 유사하게 `getStatus` 명령을 수행하면 현재 클러스터의 상태를 확인할 수 있습니다.



10. 만약 Collective Controller 를 이용해서 서버 제어를 하고 싶다면 같은 방식으로 `startCluster` 명령을 수행하면 해당 클러스터로 등록된 모든 Liberty 서버를 시작시킬 수 있습니다.



11. 또한 MBeans > WebSphere > ServerCommands > collectiveController > ServerCommands 메뉴의 리스트된 MBean 명령을 통해서 특정 서버별로 관리작업을 수행할 수도 있습니다.

12. 예를 들어 하단과 같이 cluster01 서버만 중지하는 작업을 원할 경우 해당 메뉴의 stopServer 명령을 통해서 하단과 같이 전체 클러스터가 아니라 하나의 IBM WAS Liberty 서버인 cluster01 서버만 중지/시작 할 수 있습니다.

The screenshot displays the WebSphere Administration Console interface. On the left, the 'MBeans' tree is expanded to 'ServerCommands' > 'collectiveController' > 'ServerCommands'. The 'stopServer' operation is selected. The right pane shows the configuration for the 'stopServer' operation, including parameters like 'hostName', 'wlpUserDir', 'serverName', and 'p3'. Below the configuration, the 'MBeanOperationInfo' table lists the operation details.

이름	값
조작:	stopServer
이름	stopServer
설명	Stop a registered server collective member.
영향	알 수 없음
리턴 유형	java.util.Map
매개변수-0:	hostName
이름	hostName
설명	The host name on which the target server resides. Must not be null or an empty string.
유형	java.lang.String
매개변수-1:	wlpUserDir
이름	wlpUserDir
설명	The canonical path for the user directory of server. Must not be null or an empty string.
유형	java.lang.String
매개변수-2:	serverName
이름	serverName
설명	The server name. If serverName is null, the defaultServer is assumed. Must not be an empty string.
유형	java.lang.String
매개변수-3:	p3
이름	p3

A dialog box titled '조작 리턴값' (Operation Return Value) is shown, displaying the results of the 'stopServer' operation. The 'returnCode' is 0, indicating success. The 'stdout' and 'stderr' fields show the output of the operation.

```

returnCode=0
stdout=cluster01 ¼ - ' 0, ! Á Á Æ Ç Î ' Á Á Æ Á 0 ' Î '
cluster01 ¼ - ' 0" i Á Á Æ Ç Î ¾ Á ' Î ' U.
stderr=
    
```

The bottom of the screenshot shows the 'Servers' tab with a list of WebSphere Application Servers. The 'cluster01' server is highlighted, indicating it is the target of the operation.

팁 #1) 해당 명령을 수행하기 위한 wlpUserDir 은 IBM WAS Liberty 서버의 설치 폴더\usr 디렉토리를 입력하면 됩니다. 예: C:\IBM\WebSphere85\Liberty\usr

또한 hostName 은 localhost 가 아니라 등록되어 있는 hostname 을 사용해야 합니다. Collective Controller 에 join 할 때 hostname 을 localhost 를 사용했다고 해도 실제 등록될 때는 기본 등록되어 있는 hostname 으로 변경되어서 등록되기 때문입니다.

해당 기능으로 사용하는 명령이 아니긴 하지만 이미 등록된 Host 를 대상으로 하단의 명령을 수행해보면 localhost 가 아니라 어떤 hostname 로 등록되었는지 확인할 수 있습니다.

```
collective registerHost --host=localhost --port=9443 --user=wasadm --password=wasadm
```

```
C:\IBM\WebSphere85\Liberty\bin>collective registerHost --host=localhost --port=9443 --user=wasadm --password=wasadm
호스트를 통합에 등록하는 중...
```

대상 서버와의 SSL 신뢰가 설정되지 않았습니다.

인증 체인 정보:

인증 [0]

주제 DN: CN=kr050578, OU=collective01, O=ibm, C=us

발행자 DN: OU=controllerRoot, O=00d6bc7b-0f28-4c5a-8a5f-26dd34672185, DC=com.ibm.ws.collective

일련 번호: 251,647,565,770,858

만기: 18. 10. 13 오전 11:09

SHA-1 요약: EC:D6:D8:B4:5D:97:63:85:7D:33:E1:44:5A:27:58:3E:5B:F5:E8:7F

MD5 요약: 7C:63:E9:BF:50:72:67:23:70:74:91:A7:58:ED:1A:40

인증 [1]

주제 DN: OU=controllerRoot, O=00d6bc7b-0f28-4c5a-8a5f-26dd34672185, DC=com.ibm.ws.collective

발행자 DN: OU=controllerRoot, O=00d6bc7b-0f28-4c5a-8a5f-26dd34672185, DC=com.ibm.ws.collective

일련 번호: 251,643,695,560,432

만기: 38. 10. 8 오전 11:09

SHA-1 요약: C1:A6:BD:8D:F5:EA:29:BB:87:35:A7:9D:F6:52:A5:44:EF:00:17:55

MD5 요약: 6B:ED:E3:57:3C:8A:8C:E7:59:16:31:16:72:DC:82:0C

위의 인증 체인을 승인하시겠습니까? <예/아니오>y

registerHost 를 중단합니다.

오류: kr050578 호스트가 이미 등록되어 있습니다.

4) 결론

이번 강좌에서는 IBM WAS v8.5.5 에서 강화된 Liberty 서버의 기능중의 하나로 IBM WAS Liberty 서버 통합관리를 위한 Collective Controller 를 살펴보고 테스트 해보는 시간을 가져봤습니다. 일반적인 경우에 IBM WAS ND 처럼 다수의 IBM WAS Liberty 서버를 사용하는 경우가 많지 않아서 사용될 기회가 많지는 않겠지만 그래도 만약 Cloud 환경이나 좀 더 분산된 환경을 위해서 다수의 IBM WAS Liberty 서버를 사용할 계획이 있다면 이번 강좌에서 살펴본 Collective Controller 를 이용해서 통합관리 및 제어 기능이 굉장히 편리하게 관리자 작업을 도울 수 있습니다.

이번 강좌에서는 다루지 않았지만 Collective Controller 는 이런 관리/제어 뿐만 아니라 설정 동기화, 애플리케이션 동기화등 보다 다양한 기능을 제공하기 때문에 해당 기능들은 기회가 되는데로 다른 강좌에서 좀 더 자세히 다뤄보도록 하겠습니다.

그럼 여기서 이만 이번 강좌는 마무리하고 다음 강좌에서 뵙도록 하겠습니다. Go Go !!!

9) 참고 자료

1. 이 가이드는 IBM WAS v8.5.5 최초 사용자를 위한 기본 가이드 입니다.
2. IBM WAS 자체에 아직 익숙하지 않으신 분들은 가급적 기본강좌인 '하나씩 쉽게 따라 해보는 IBM WAS v7' 강좌와 '제대로 맛보는 IBM WAS v8.5' 강좌를 먼저 읽고 이 강좌를 읽으시는 것이 이해에 훨씬 도움이 됩니다.
(http://www.websphere.pe.kr/xe/?mid=was_info_re&page=3&document_srl=800
http://www.websphere.pe.kr/xe/?mid=was_info_re&page=2&document_srl=134863)
3. 가급적 IBM WAS v8.5 InfoCenter 의 해당 카테고리를 한 번 읽어보고 난 후에 작업하시기 바랍니다.
4. InfoCenter – WebSphere Application Server v8.5
(<http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/index.jsp>)
5. InfoCenter – Liberty profile: Collective architecture
(http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wlp.nd.doc/ae/cwlp_collective_arch.html?resultof=%22%63%6f%6c%6c%65%63%74%69%76%65%22%20%22%63%6f%6e%74%72%6f%6c%6c%65%72%22%20)