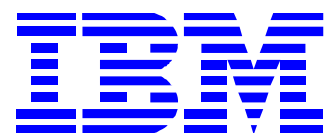


WebSphere Application Server v8.5.5
IBM WAS Base 버전과 WXS 를 이용한
Session Cluster 구축 2

(2014. 03.)

IBM SWG WebSphere Technical Sales

이정운 과장(juwlee@kr.ibm.com)



0) IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축 간략 소개

안녕하세요 freeman 입니다.

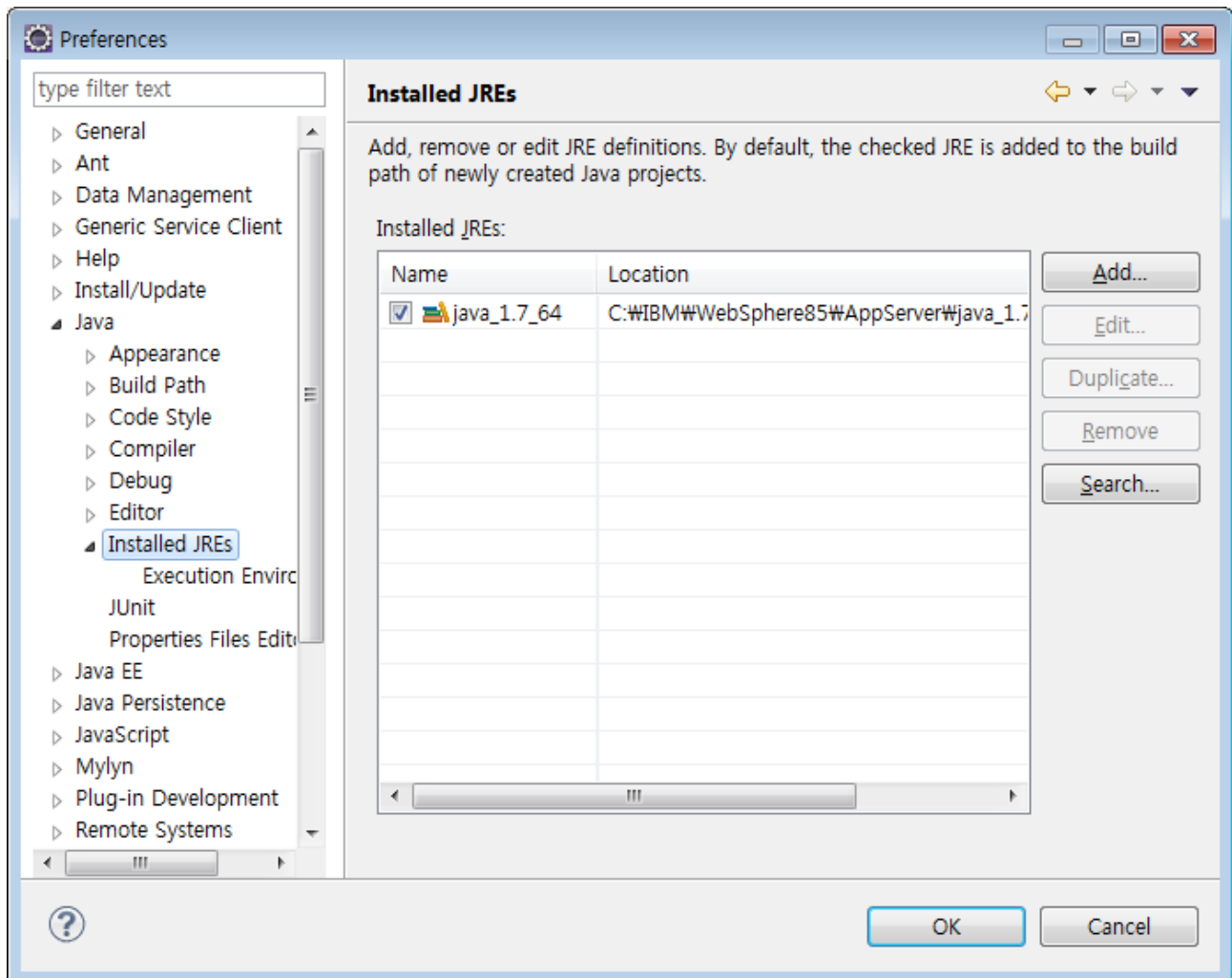
이번 강좌에서 언급드릴 내용은 IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축 그 두 번째 이야기 입니다. 이미 지난번의 "[WAS8.5.5]하나씩 이해하는 IBM WAS v8.5.5 - 4. IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축 (http://www.websphere.pe.kr/xe/was_info_re/174609)" 강좌에서 IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축에 대한 이야기를 진행했지만 좀 더 쉬운 방법을 요구하거나 튜닝에 대한 문의와 같은 여러 요구 사항이 있어서 좀 더 자세한 이야기를 강좌로 진행하고자 합니다.

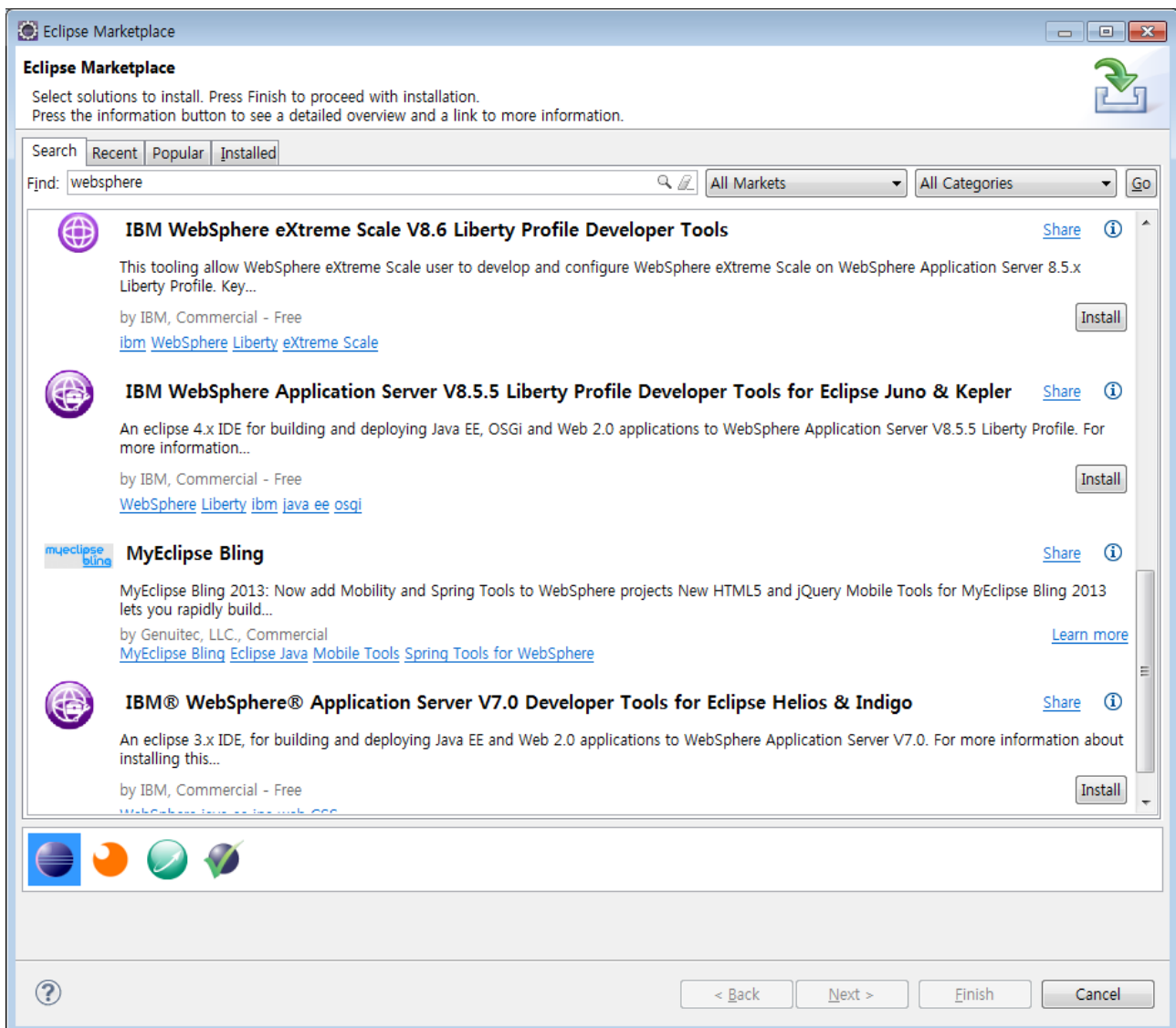
이번 강좌에서는 IBM WAS base 의 AdminAgent 를 이용해서 WAS 뿐만 아니라 WXS 까지 통합 관리 콘솔에서 관리가 가능한 형태로 구성하는 방법과 이전 강좌에서는 조금 소홀히 했던 WXS 설정에 대한 대략적인 설명등을 좀 더 자세히 이야기 해보도록 하겠습니다.

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

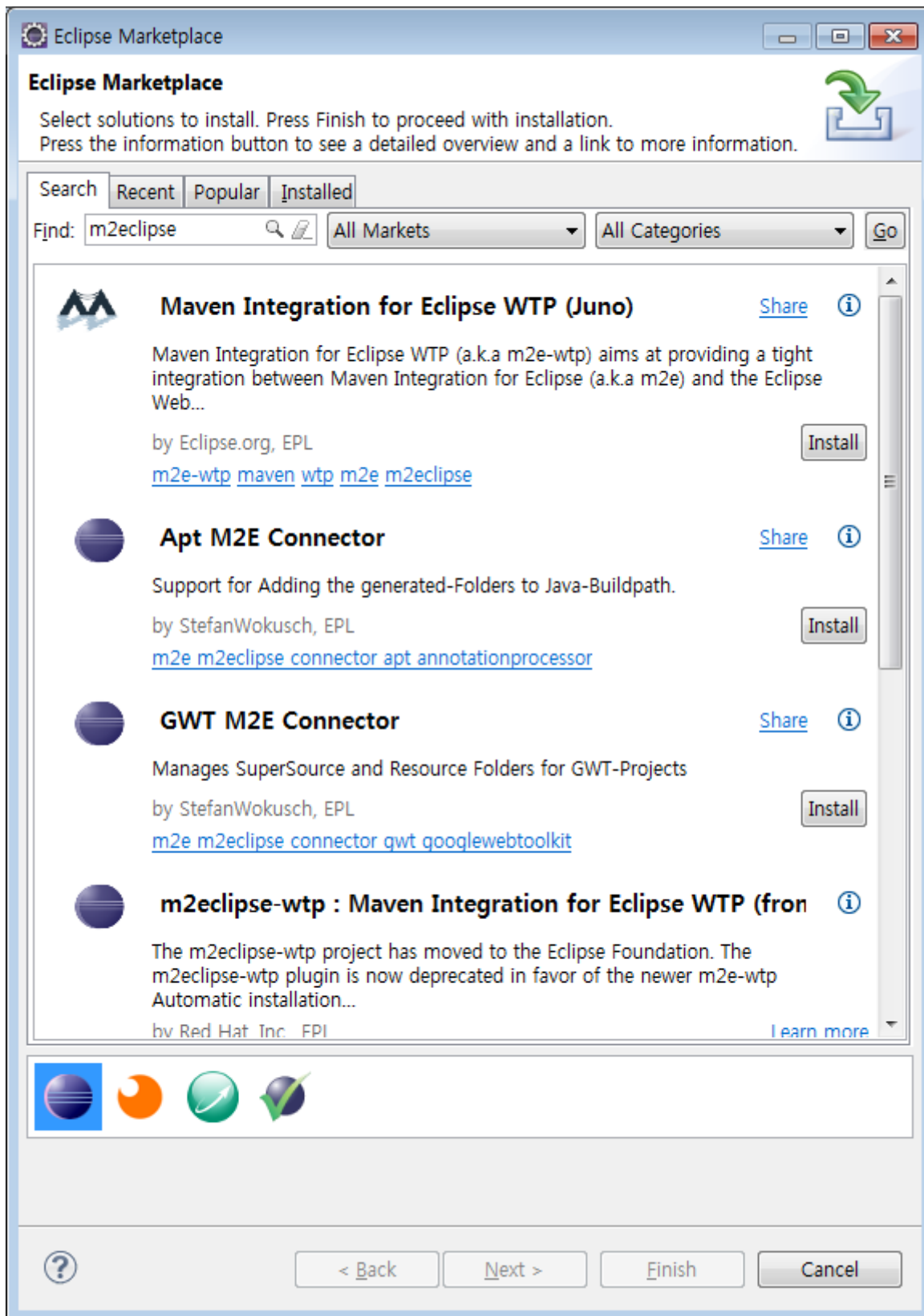
1) AdminAgent 를 이용한 IBM WAS & WXS 통합 관리 - 설정하기

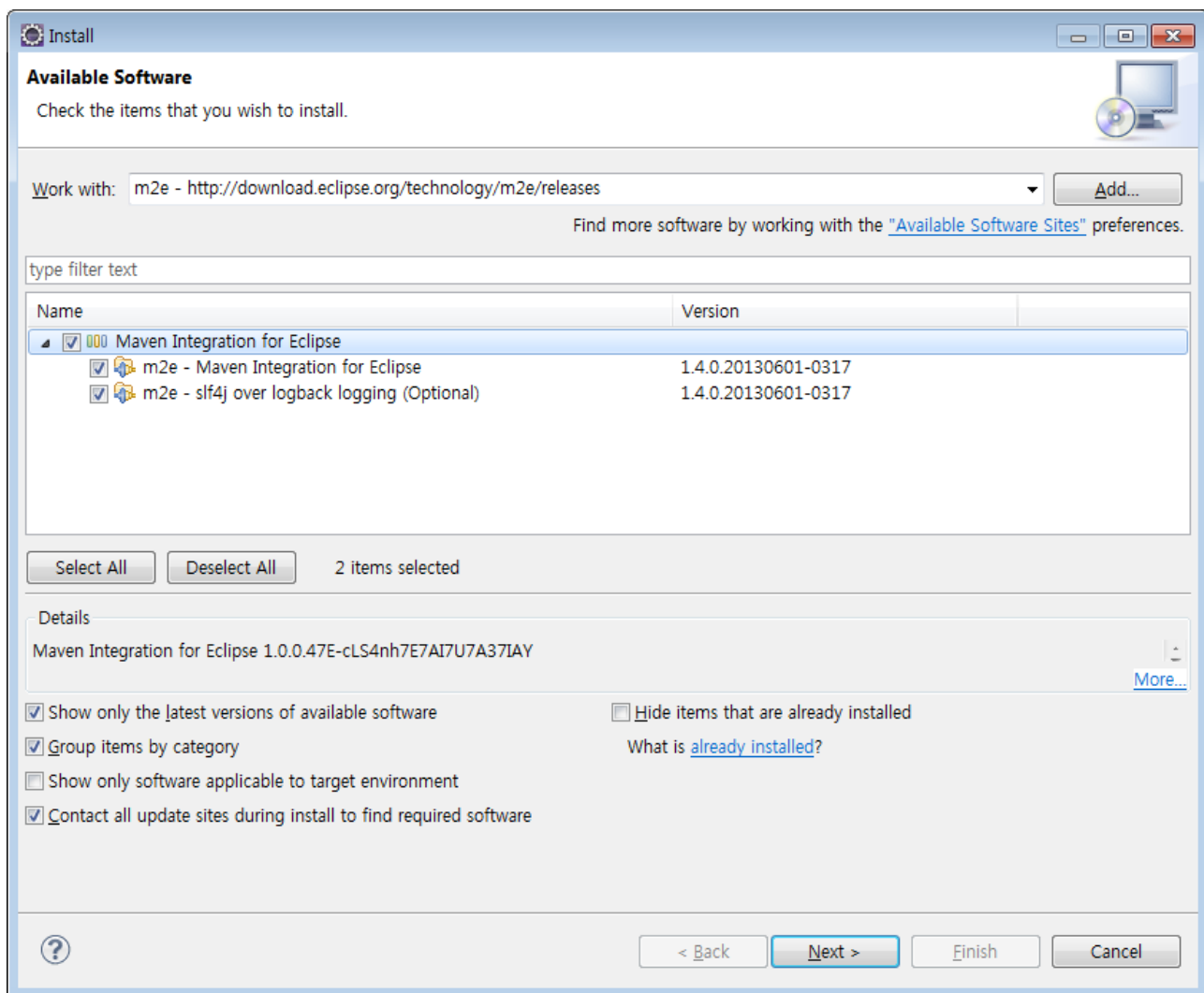
1. IBM WAS Base 버전에서는 서버내의 통합관리를 위하여 AdminAgent 를 제공합니다. (참고: [WAS8.5.5] 하나씩 이해하는 IBM WAS v8.5.5 - 5. IBM WAS Base 에서 Admin Agent 를 통한 관리 설정 (http://www.websphere.pe.kr/xe/was_info_re/186253)) 이를 활용하면 이전에 설명드린 강좌에서와 다르게 WAS 뿐만 아니라 WXS 까지 관리콘솔 형태로 통합관리/운영이 가능합니다.
2. 본 강좌에서는 AdminAgent 를 위해 만들어진 AdminAgent01 프로파일과 WAS server 용으로 사용할 AppSrv01 프로파일이 만들어져 있다는 것과 IBM WAS base 에 WXS 가 같이 설치된 것을 가정하고 진행하도록 하겠습니다. 해당 부분이 이해가 잘 안되는 분들은 이전 강좌들을 다시 한번 참고해 주시기 바라겠습니다. (해당 강좌를 진행하기 위해서는 반드시 IBM WXS v8.6.0.5 이상의 fixpack 이 설치되어 있어야 합니다.)
3. IBM WAS 와 WXS 를 연동하기 위해서는 제일 첫 스텝으로 이미 만들어진 WAS 프로파일을 WXS 용으로 강화해야 합니다.(IBM WAS 가 설치된 폴더의 bin 디렉토리의 ProfileManagement 폴더에 가서 pmt.bat 을 클릭하여 WebSphere Customization Toolkit 을 수행한 후 기능 보강을 클릭하면 됩니다.)

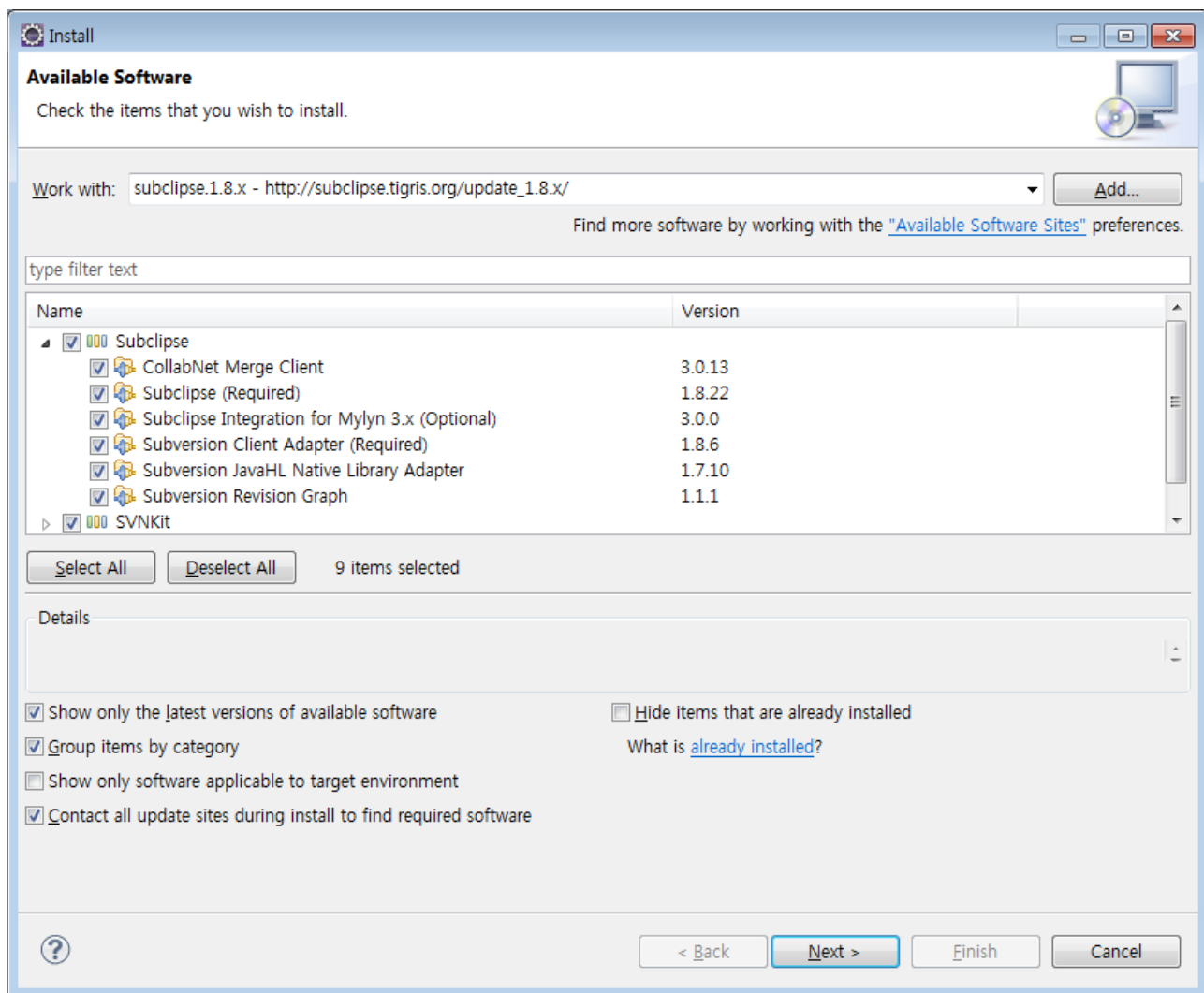




```
C:\Users\IBM_ADMIN>mvn --version
Apache Maven 3.2.1 (ea8b2b07643dbb1b84b6d16e1f08391b666bc1e9; 2014-02-15T02:37:52+09:00)
Maven home: D:\open_source\apache-maven-3.2.1
Java version: 1.7.0, vendor: IBM Corporation
Java home: C:\IBM\WebSphere85\AppServer\java_1.7_64\jre
Default locale: ko_KR, platform encoding: MS949
OS name: "windows 7", version: "6.1", arch: "amd64", family: "windows"
C:\Users\IBM_ADMIN>
```







5. 통합 관리작업 설정을 위하여 하단과 같이 adminagent 를 시작시킵니다.

startServer adminagent

```
C:\IBM\WebSphere85Base\AppServer\profiles\AdminAgent01\bin>startServer adminagent
ADMU0116I: 도구 정보가
C:\IBM\WebSphere85Base\AppServer\profiles\AdminAgent01\logs\adminagent\startServer.log
파일에 로그 중입니다.
ADMU0128I: AdminAgent01 프로파일로 도구 시작 중
ADMU3100I: 다음 서버에 대한 구성을 읽는 중: adminagent
ADMU3200I: 서버가 실행되었습니다. 초기화 상태 대기 중입니다.
ADMU3000I: adminagent 서버가 e-business 용으로 열렸습니다. 프로세스 ID: 34224

C:\IBM\WebSphere85Base\AppServer\profiles\AdminAgent01\bin>
```

6. adminagent 에 만들어 둔 AppSrv01 프로파일을 연동하기 위해 registerNode 명령을 수행합니다.
(adminagent 의 SOAP 포트는 사전에 파악한후 script 에 넣어주어야 합니다.)

registerNode -conntype SOAP -port 8877 -profilePath

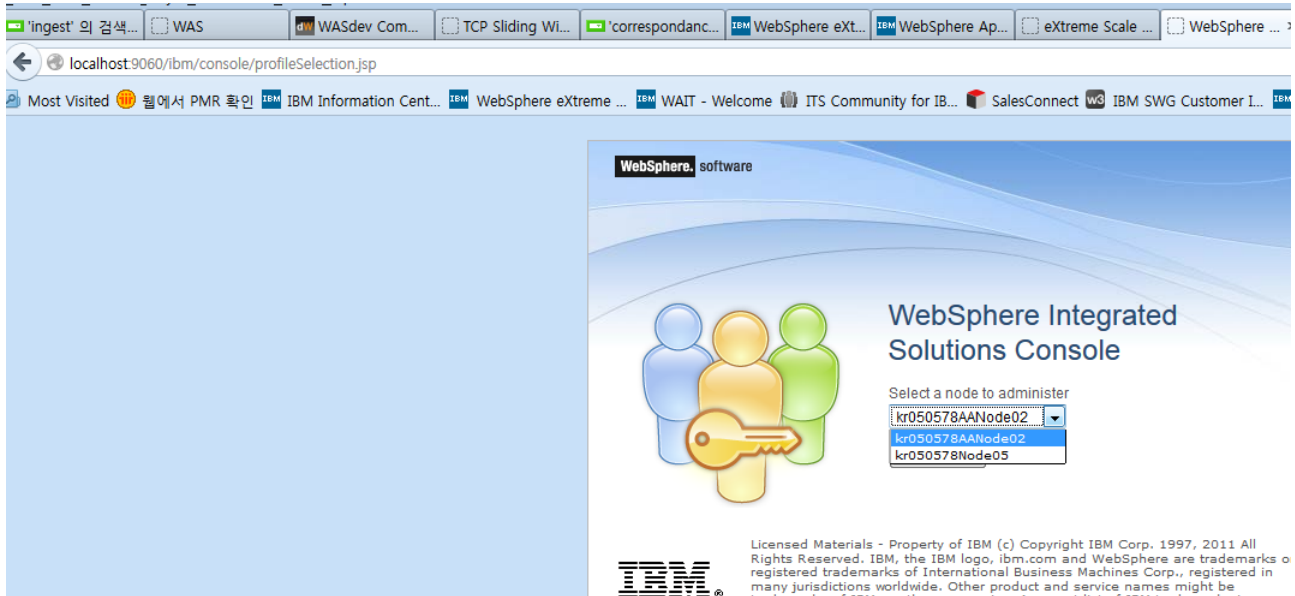
C:\IBM\WebSphere85Base\AppServer\profiles\AppSrv01

```
C:\IBM\WebSphere85Base\AppServer\profiles\AdminAgent01\bin>registerNode -conntype SOAP -port 8877 -profilePath C:\IBM\WebSphere85Base\AppServer\profiles\AppSrv01
ADMU0116I: 도구 정보가
C:\IBM\WebSphere85Base\AppServer\profiles\AdminAgent01\logs\registerNode.log
파일에 로그 중입니다.
ADMU0128I: AdminAgent01 프로파일로 도구 시작 중
ADMU8053I: AdminAgent 서버에 연결됨: localhost:8877
ADMU8002I: 경로가 C:\IBM\WebSphere85Base\AppServer\profiles\AppSrv01 인 노드와 adminagent 간의 서명자 교환 중.
ADMU8007I: 서명자가 교환되었습니다.
ADMU0505I: 구성에서 서버를 찾았습니다.
ADMU0506I: 서버 이름: server1
ADMU2010I: kr050578Node05 노드에 대한 모든 서버 프로세스를 중지하고 있습니다.
ADMU0512I: server1 서버에 도달할 수 없습니다. 중지된 것으로 보입니다.
ADMU8010I: 경로가 C:\IBM\WebSphere85Base\AppServer\profiles\AppSrv01 인 애플리케이션 서버의 등록을 시작합니다.
ADMU0024I: 이전 백업 디렉토리를 삭제하는 중입니다.
ADMU8004I: 등록될 노드의 원래 config 디렉토리를 백업 중입니다.
ADMU8037I: 등록될 노드의 원래 wsadmin.properties 파일을 백업 중입니다.
ADMU8036I: 노드를 AdminAgent 에 등록합니다.
ADMU8042I: 노드가 등록되었습니다.
ADMU8040I: 관리 에이전트가 등록된 노드의 관리 서브시스템을 초기화하는 중입니다.
ADMU8014I: 등록된 노드의 관리 서브시스템이 초기화되었습니다.
ADMU8041I: 관리 에이전트가 등록된 노드의 관리 서브시스템을 시작하는 중입니다.
ADMU8015I: 등록된 노드의 관리 서브시스템이 시작되었습니다.
ADMU0505I: 구성에서 서버를 찾았습니다.
ADMU0506I: 서버 이름: server1
ADMU7703I: 현재 server1 과<와> 연관된 Windows 서비스 kr050578Node05 이<가> 등록 취소되고 있습니다.
ADMU8012I: 경로가 C:\IBM\WebSphere85Base\AppServer\profiles\AppSrv01 인 애플리케이션 서버가 등록되었습니다.

C:\IBM\WebSphere85Base\AppServer\profiles\AdminAgent01\bin>
```


7. adminagent 를 수행한 후 브라우저를 이용하여 관리콘솔에 접속하면 상단에서 수행한 registerNode 명령을 통해 AppSrv01 프로파일이 정상적으로 연결된 것을 확인하실 수 있으며(Node 명으로 확인가능) 해당 노드로 접속합니다.

<http://localhost:9060>(관리포트 번호)/ibm/console



8. Session Cluster 구성/관리 테스트를 위해서 총 4개의 server 를 하단과 같이 생성합니다.

catalogServer01 : WXS 의 카탈로그 서버 용도로 활용할 WAS 인스턴스

-> 카탈로그 서버란 일종의 WXS 의 마스터 서버로써 컨테이너 서버의 파티션 관리 및 어떤 파티션에 실제 요청하는 데이터가 있는지 정보들을 가지고 있는 WXS 관리 서버입니다. WXS 구성을 사용할 경우 모든 요청은 카탈로그 서버에서 최초 index 정보를 가지고오며 이후는 직접 컨테이너 서버에 접속하여 캐쉬된 데이터를 가지고 올 수 있습니다. 필요시 HA 구성이 가능합니다.

containerServer01, containerServer02 : WXS 의 컨테이너 서버 용도로 활용할 WAS 인스턴스

-> 컨테이너 서버란 말 그대로 캐쉬된 데이터를 저장하는 용도로 사용하는 WXS 서버 입니다. 캐쉬를 얼마나 저장할 지 요구량에 따라서 원하시는 만큼의 병렬 구성이 가능합니다.

server1, server2 : Session cluster 테스트를 위해 사용할 테스트 WAS 인스턴스

Select	Name	Node	Host Name	Version	Status
You can administer the following resources:					
<input type="checkbox"/>	catalogServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	✖
<input type="checkbox"/>	containerServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	✖
<input type="checkbox"/>	containerServer02	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	✖
<input type="checkbox"/>	server1	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	✖
<input type="checkbox"/>	server2	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	✖

참고: 향후 WXS 카탈로그 서버와 컨테이너 서버의 시작시에 Host name 형태의 lookup 이 이루어지므

로 개인 컴퓨터에서 테스트 하시는 분들은 hosts 파일에 Host name 을 등록해 두어야 합니다.

팁#1) 카탈로그 서버나 컨테이너 서버의 경우 WAS 설정의 "Start components as needed (필요한 시작 컴포넌트)" 옵션을 하단과 같이 enable 하는 것을 권고 드립니다. 해당 옵션을 사용하게 되면 WAS 가 구동시 필요한 컨테이너만 동적 로딩하게 되어 있으므로 현재 설정상 불필요한 "웹 서비스 컨테이너", "SIP 컨테이너" 등이 구동되지 않습니다. 이를 통해서 메모리 풋프린트를 줄일 수 있으며 시작 시간을 빠르게 가지고 갈 수 있습니다.

Application servers > catalogServer01
Use this page to configure an application server. An application server is a server

Configuration

General Properties

Name
catalogServer01

Node name
kr050578Node05

☐ Run in development mode

☒ Parallel start

☒ Start components as needed

9. 카탈로그 서버 설정을 위해서 관리콘솔에서 System administration > WebSphere eXtreme Scale > Catalog service domains 메뉴를 선택하고 New 를 클릭합니다.

localhost:9060/ibm/console/login.do

Most Visited 웹에서 PMR 확인 IBM Information Cent... WebSphere eXtreme ... WAIT - Welcome ITS Community for IB...

WebSphere. software

View: All tasks

- Welcome
- Guided Activities
- Servers
- Applications
- Services
- Resources
- Security
- Environment
- System administration
 - Extended Repository Service
 - Save changes to master repository
 - WebSphere eXtreme Scale**
 - Catalog service domains

Cell=kr050578Node03Cell, Profile=AppSrv01

Catalog service domains

A WebSphere eXtreme Scale catalog service domain is a highly available collection of catalog server Application Server server processes within a single cell and core group. The catalog service domain c WebSphere Application Server cells or as Java SE processes. The catalog service controls the placem container servers in the data grid.

Preferences

New... Delete Test connection Set default

Select Name Default

None

Total 0

10. 카탈로그 도메인 이름을 입력하고 카탈로그 서버로 지정할 서버 선택후에 Client port 를 입력하고 'OK' 를 클릭하면 카탈로그 도메인이 생성됩니다.

Catalog service domains > New...

A WebSphere eXtreme Scale catalog service domain is a highly available collection of catalog servers. This collection of catalog servers can run in WebSphere Application Server server processes within a single cell and core group. The catalog service domain can also define a group of remote servers that run in different WebSphere Application Server cells or as Java SE processes. The catalog service controls the placement of shards and discovers and monitors the health of the container servers in the data grid.

General Properties

* Name
CatalogDomain01

☒ Enable this catalog service domain as the default unless another catalog service domain is explicitly specified.

☒ Enable IBM eXtremeIO (XIO) communication. This configuration option is not applicable for domains that are configured with remote server members.

The additional properties will not be available until the general properties for this item are applied or saved.

Additional Properties

- ☐ Client security properties
- ☐ Custom properties

Catalog Servers

New Delete

Select	Catalog Server Endpoint	Client Port	Listener Port
<input checked="" type="radio"/>	Existing application server kr050578Node03Cell\kr050578Node05\catalogServer01	6600	
<input type="radio"/>	Remote server		

Apply OK Reset Cancel

11. 이미 만들어 놓은 카탈로그 서버를 시작시킨 후에 생성된 카탈로그 도메인에서 Test Connection 을 통해서 문제없이 카탈로그 서버와 접속이 가능한지 확인합니다.

elect	Name	Node	Host Name	Version	Status
You can administer the following resources:					
<input type="checkbox"/>	catalogServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	

Messages

Connection successful for catalog service CatalogDomain01: started.

Catalog service domains

A WebSphere eXtreme Scale catalog service domain is a highly available collection of catalog servers. This collection of catalog servers can run in WebSphere Application Server server processes within a single cell and core group. The catalog service domain can also define a group of remote servers that run in different WebSphere Application Server cells or as Java SE processes. The catalog service controls the placement of shards and discovers and monitors the health of the container servers in the data grid.

Preferences

New... Delete Test connection Set default

Select Name Default

You can administer the following resources:

<input type="checkbox"/>	CatalogDomain01	
--------------------------	-----------------	--

Total 1

참고 : 카탈로그 서버의 일반적인 listener port 는 XIO_Address 포트를 사용합니다. 해당 포트를 모르겠 으면 관리콘솔에서 Server > 해당 서버 이름 > port 를 클릭하여 확인할 수 있습니다.

팁1) 만약 카탈로그 서버를 다른 HW 서버에 이중화 구성한다고 하면 하단과 같이 방금 생성한 카탈로그 서비스 도메인에서 'New' 를 클릭하여 추가해주기만 하면 됩니다.

Catalog service domains > CatalogDomain01

A WebSphere eXtreme Scale catalog service domain is a highly available collection of catalog servers. This collection of catalog servers can run server processes within a single cell and core group. The catalog service domain can also define a group of remote servers that run Application Server cells or as Java SE processes. The catalog service controls the placement of shards and discovers and monitors the health of the data grid.

General Properties

* Name

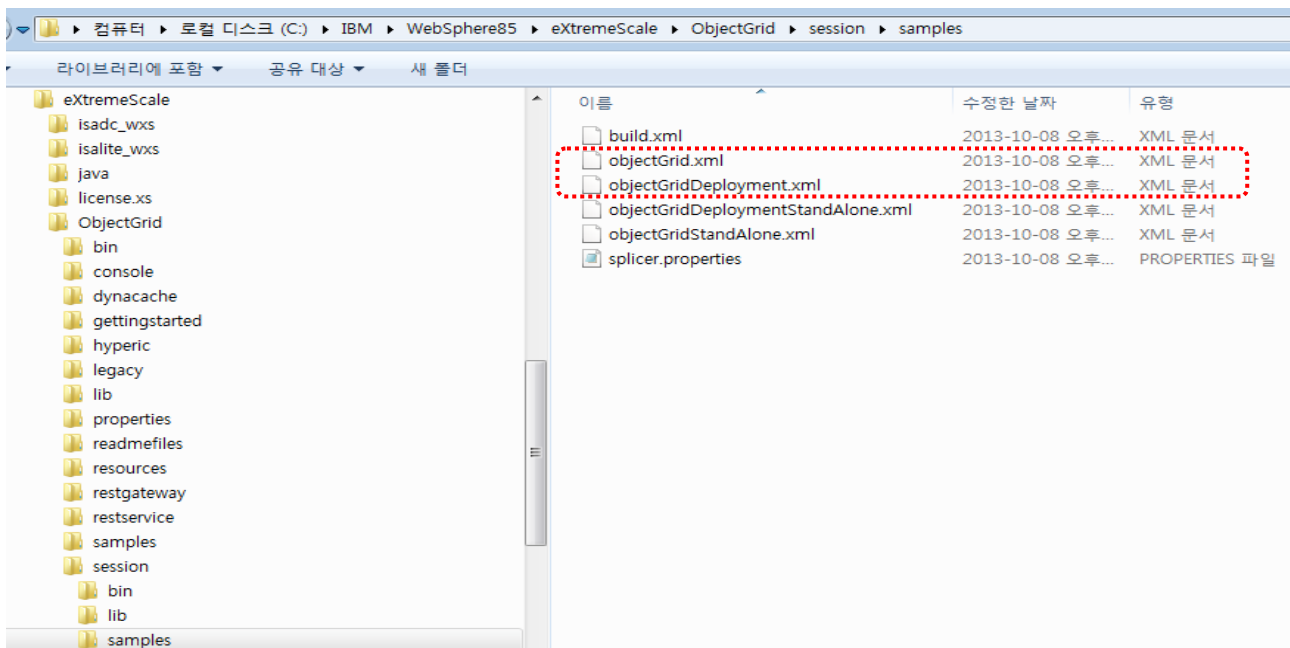
☒ Enable this catalog service domain as the default unless another catalog service domain is explicitly specified.

☒ Enable IBM eXtremeIO (XIO) communication. This configuration option is not applicable for domains that are configured with remote server members.

Catalog Servers

Select	Catalog Server Endpoint	Client Port	Listener Port	Status
<input type="checkbox"/>	kr050578Node03Cell\kr050578Node05\catalogServer01	6600		✖
<input type="checkbox"/>	<input type="radio"/> Existing application server <input type="text" value="kr050578Node03Cell\kr050578Node05\containerServer01"/>	<input type="text"/>	<input type="text" value="4811"/>	
<input type="checkbox"/>	<input checked="" type="radio"/> Remote server <input type="text" value="another server hostname"/>			

12. 카탈로그 서버가 문제없이 구동되었으니 컨테이너 서버에 대한 설정을 해야할 차례입니다. 컨테이너 서버는 내부 구성을 어떻게 할지에 대해 objectGrid.xml, objectGridDeployment.xml 이라는 두가지 설정 파일을 통해서 수행합니다. objectGridDeployment.xml 은 컨테이너 서버 안에 어떻게 Object Grid 를 구성할 것인가(파티션 개수, 복제본 개수 등) 에 대한 설정이며 objectGrid.xml 은 그렇게 구성된 Object Grid 에 대한 세부 설정(backingMap 정보, lock 방식 등) 에 관련된 파일입니다. 기본적으로 Session 저장 위한 설정 부분은 WXS 제품 설치 디렉토리에 session > sample 디렉토리에 하단과 같이 이미 완성해 놓은 버전을 같이 제공하고 있으므로 해당 설정을 사용하면 됩니다.



참고 : WXS 8.6 InfoCenter - ObjectGrid descriptor XML file

<http://pic.dhe.ibm.com/infocenter/wxinfo/v8r6/topic/com.ibm.websphere.extremescale.doc/rxsogref.html>

objectGrid.xml 샘플

```
<?xml version="1.0" encoding="UTF-8"?>
<objectGridConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://ibm.com/ws/objectgrid/config ../objectGrid.xsd"
xmlns="http://ibm.com/ws/objectgrid/config">
  <objectGrids>
    <objectGrid name="session" txTimeout="30">
      <bean id="ObjectGridEventListener"
className="com.ibm.ws.xs.sessionmanager.SessionHandleManager"/>
      <backingMap name="objectgridSessionMetadata"
pluginCollectionRef="objectgridSessionMetadata" readOnly="false" lockStrategy="PESSIMISTIC"
ttlEvictorType="LAST_ACCESS_TIME" timeToLive="3600" copyMode="NO_COPY"/>
      <backingMap name="objectgridSessionAttribute.*" template="true" readOnly="false"
lockStrategy="PESSIMISTIC" ttlEvictorType="NONE" copyMode="NO_COPY"/>
      <backingMap name="objectgridSessionTTL.*" template="true" readOnly="false" lockStrategy="PESSIMISTIC"
ttlEvictorType="LAST_ACCESS_TIME" timeToLive="3600" copyMode="NO_COPY"/>
    </objectGrid>
  </objectGrids>
  <backingMapPluginCollections>
    <backingMapPluginCollection id="objectgridSessionMetadata">
      <bean id="MapEventListener" className="com.ibm.ws.xs.sessionmanager.MetadataMapListener"/>
    </backingMapPluginCollection>
  </backingMapPluginCollections>
</objectGridConfig>
```

txTimeout : 트랜잭션 타임아웃 값, 트랜잭션을 완료하는데 해당 시간 보다 오래 걸린다면 TransactionTimeoutException 을 반환하고 해당 트랜잭션은 롤백 됩니다.

readOnly : backingMap 인스턴스 읽기/쓰기 모드 설정 값, true 로 변경하면 해당 backingMap 은 오직 읽기만 가능합니다.

lockStrategy : 트랜잭션에 의해 map 엔트리에 접근할 때 마다 internal lock manager 가 사용되며 3가지 전략을 제공합니다. - OPTIMISTIC, PESSIMISTIC, NONE

- Pessimistic : Data 가 자주 변하는 경우에 사용되며 cache 엔트리를 read 할 때 lock 을 얻어오고 조건부로 트랜잭션이 완료될 때까지 lock 을 유지합니다. Lock 을 유지하는 시간은 Transaction isolation level 튜닝을 통해 조절 가능합니다.
- Optimistic : pessimistic 에 비해서 성능이나 확장성이 더 향상된 locking 전략이며 애플리케이션이

때때로 발생하는 update 실패에 대해서 큰 이슈가 없다면 Optimistic 전략 보다 좋은 성능을 보장합니다. 주로 read 가 많으며 write 나 update 가 간헐적인 경우에 적합한 lock 전략입니다.

- None : internal lock manager 가 필요없을 경우에 사용되며 동시성 제어는 eXtreme Scale 외부에서 이루어져야 합니다..

ttlEvictorType : BackingMap 엔트리에 대한 만료 시간을 어떻게 계산할 것인지에 대한 옵션 값이며 CREATION_TIME, LAST_ACCESS_TIME, LAST_UPDATE_TIME, NONE 중에 선택이 가능합니다.

timeToLive : 여기서 지정된 값을 기반으로 TTL evictor 가 map 엔트리를 메모리에서 퇴출시킵니다.

copyMode : get 명령시에 BackingMap 인스턴스가 실제 값을 어떻게 반환할지에 대한 옵션 값입니다.

- COPY_ON_READ_AND_COMMIT : 애플리케이션이 BackingMap 인스턴스의 value object 에 대한 참조를 가지고 있지 않는 것을 보장합니다. 대신에, 애플리케이션은 항상 BackingMap instance 값의 복제본을 가지고 작업합니다.
- COPY_ON_READ : 트랜잭션이 완료되었을 때 발생하는 복제본을 제거함으로써 COPY_ON_READ_AND_COMMIT 보다 성능이 더 좋습니다. BackingMap data 의 무결성을 보존하기 위하여 애플리케이션은 트랜잭션이 완료된 후 엔트리에 대한 모든 참조를 지우는 작업을 수행합니다. 해당 옵션을 사용하면 ObjectMap.get 메소드 수행시 트랜잭션이 완료되기 전까지 애플리케이션의 변경 작업이 BackingMap 요소에 영향을 미치지 못하도록 value 에 대한 참조가 아니라 복사본을 반환합니다.
- COPY_ON_WRITE : ObjectMap.get 메소드 수행시 value object 에 대한 직접 참조 대신에 value 에 대한 proxy 를 반환합니다. Proxy 는 애플리케이션에 의해서 value interface 의 set 메소드가 호출되지 않는다면 value 의 복제본을 만들지 않는 것을 보장합니다.
- NO_COPY : 성능 향상을 위하여 ObjectMap.get 메소드를 통해 가지고 온 value object 가 절대 변경되지 않게 합니다.
- COPY_TO_BYTES : 복잡한 object 타입의 메모리 풋 프린트를 줄이고 복제본을 만들기 위하여 직렬화가 필요한 객체의 복제 성능을 향상하기 위해 사용됩니다.

참고 : WXS 8.6 InfoCenter - Deployment policy descriptor XML file

[http://pic.dhe.ibm.com/infocenter/wxsinfo/v8r6/topic/](http://pic.dhe.ibm.com/infocenter/wxsinfo/v8r6/topic/com.ibm.websphere.extremescale.doc/rxsdplcyref.html)

[com.ibm.websphere.extremescale.doc/rxsdplcyref.html](http://pic.dhe.ibm.com/infocenter/wxsinfo/v8r6/topic/com.ibm.websphere.extremescale.doc/rxsdplcyref.html)

objectGridDeployment.xml 샘플

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<deploymentPolicy xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
    xsi:schemaLocation="http://ibm.com/ws/objectgrid/deploymentPolicy ../deploymentPolicy.xsd"
```

```
    xmlns="http://ibm.com/ws/objectgrid/deploymentPolicy">
```

```
    <objectgridDeployment objectgridName="session">
```

```
        <mapSet name="sessionMapSet" numberOfPartitions="5" minSyncReplicas="0" maxSyncReplicas="0"
```

```
        maxAsyncReplicas="1" developmentMode="true" placementStrategy="PER_CONTAINER">
```

```
        <map ref="objectgridSessionMetadata"/>
```

```

<map ref="objectgridSessionAttribute.*"/>

<map ref="objectgridSessionTTL.*"/>

    </mapSet>

</objectgridDeployment>

</deploymentPolicy>

```

numberOfPartitions : mapSet 요소의 파티션 개수 지정

minSyncReplicas : 각 파티션에 대한 동기식 복제본의 최소 개수 지정

maxSyncReplicas : 각 파티션에 대한 동기식 복제본의 최대 개수 지정

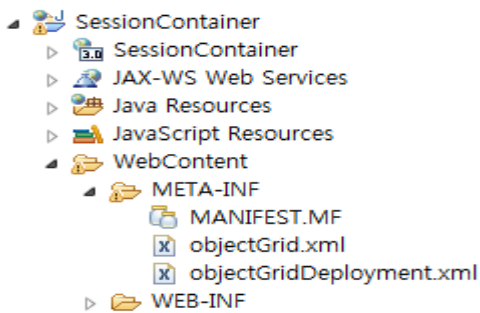
maxAsyncReplicas : 각 파티션에 대한 비동기식 복제본의 최대 개수 지정

developmentMode : 개발 모드 사용 여부에 대한 옵션(true/false), true 일 경우에는 하나의 컴퓨터에 원본 파티션과 복제본 파티션이 동시에 존재할 수 있음

replicaReadEnabled : read 요청에 대해서 primary 파티션 뿐만 아니라 복제본에서도 처리할 수 있도록 지정하는 옵션

placementStrategy : FIXED_PARTITION - 가용한 컨테이너 서버들에 primary 샤드가 지정된 수치 만큼 배치됨, PER_CONTAINER - 컨테이너 서버별로 지정된 수치 만큼 primary 샤드가 배치됨

13. 컨테이너 서버에 대한 설정을 하기 위하여 하단과 같이 eclipse 와 같은 개발 도구를 통하여 빈 WAR 파일을 하나 만들고 META-INF 디렉토리에 위에서 언급했던 objectGrid.xml, objectGridDeployment.xml 설정 파일을 복사해 놓습니다. 이 후 해당 WAR 파일을 컨테이너 서버(containerServer01, containerServer02) 에 배포한 후 해당 서버를 시작시키면 동적으로 objectGrid.xml, objectGridDeployment.xml 설정 파일을 인식하여 컨테이너 서버 설정을 완료합니다. (두 개의 설정 파일의 이름은 변경되면 안됩니다.)



Enterprise Applications > SessionContainer_war > Manage Modules

Manage Modules

Specify targets such as application servers or clusters of application servers where you want to install the modules that are contained in your application. Modules can be installed on the same application server or dispersed among several application servers. Also, specify the Web servers as targets that serve as routers for requests to this application. The plug-in configuration file (plugin-cfg.xml) for each Web server is generated, based on the applications that are routed through.

Clusters and servers:

```

WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=server1
WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=catalogServer01
WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=containerServer01
WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=server2
WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=containerServer02

```

Apply








Remove Update Remove File Export File				
Select	Module	URI	Module Type	Server
<input type="checkbox"/>	SessionContainer_war	SessionContainer_war;WEB-INF/web.xml	Web Module	WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=containerServer02 WebSphere:cell=kr050578Node03Cell,node=kr050578Node05,server=containerServer01

14. 카탈로그 서버와 컨테이너 서버를 하나 구동합니다.

Application servers

Use this page to view a list of the application servers in your environment and the status of each of these servers. You can also use this page to change the status of a specific application server.

Preferences

New...	Delete	Templates...	Start	Stop	Restart	ImmediateStop	Terminate
<div></div>							
Select	Name ▾	Node ▾	Host Name ▾	Version ▾	Status 		
You can administer the following resources:							
<input type="checkbox"/>	catalogServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4			
<input type="checkbox"/>	containerServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4			

15. xscmd 명령을 통해서 파티션 구성을 확인합니다. objectGridDeployment.xml 설정파일에 지정된 것처럼 5개의 파티션이 생성된 것을 하단과 같이 확인할 수 있습니다.

xscmd -c showMapSizes -cep kr050578(호스트명):4811(XIO_ADDRESS 포트)

```
C:\IBM\WebSphere85\ExtremeScale\ObjectGrid\bin>xscmd -c showMapSizes -cep kr050578:4811
시작 시간 :2014-03-06 14:26:44.551

CWXSI0068I: 명령 실행 중: showMapSizes

*** session 데이터 그리드 및 sessionMapSet 맵 세트에 대한 결과 표시.

*** kr050578Node03Cell\kr050578Node05\containerServer01에 대한 맵 나열 ***
맵 이름                파티션 맵 항목 사용한 바이트 샤드 유형 컨테이너
-----
objectgridSessionMetadata 0      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 1      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 2      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 3      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 4      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
서버 총계: 0 <0 B>

총 카탈로그 서비스 도메인 수: 0 <0 B>
<사용된 바이트 통계는 단순 오브젝트 또는 COPY_TO_BYTES 복사 모드를 사용 중인 경우에만 정확합니다.>

CWXSI0040I: showMapSizes 명령이 완료되었습니다.

종료 시간: 2014-03-06 14:26:47.766

C:\IBM\WebSphere85\ExtremeScale\ObjectGrid\bin>
```

16. 현재 상태에서 containerServer02 를 추가적으로 시작시킵니다.

Select	Name	Node	Host Name	Version	Status
You can administer the following resources:					
<input type="checkbox"/>	catalogServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	
<input type="checkbox"/>	containerServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	
<input type="checkbox"/>	containerServer02	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	

17. 다시한번 xscmd 명령을 통해서 파티션 구성을 확인합니다. 하단의 결과 화면에서 확인하실 수 있는 것처럼 컨테이너 서버가 하나가 더 구동되면서 나머지 한쪽에는 비동기 복제(AsynchronousReplica) 파티션이 위치되는 것을 확인할 수 있습니다.

```
C:\IBM\WebSphere85\extremeScale\ObjectGrid\bin>xscmd -c showMapSizes -cep kr050578:4811
시작 시간 :2014-03-06 14:28:59.059

CWXS10068I: 명령 실행 중: showMapSizes

*** session 데이터 그리드 및 sessionMapSet 맵 세트에 대한 결과 표시.

*** kr050578Node03Cell\kr050578Node05\containerServer02에 대한 맵 나열 ***
맵 이름                파티션 맵 항목 사용한 바이트 샤드 유형      컨테이너
이름
-----
objectgridSessionMetadata 0      0      0      AsynchronousReplica kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 1      0      0      AsynchronousReplica kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 2      0      0      AsynchronousReplica kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 3      0      0      AsynchronousReplica kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 4      0      0      AsynchronousReplica kr050578Node03Cell\kr050578Node05\containerServer02_C-2
서버 총계: 0 <0 B>

*** kr050578Node03Cell\kr050578Node05\containerServer01에 대한 맵 나열 ***
맵 이름                파티션 맵 항목 사용한 바이트 샤드 유형      컨테이너
이름
-----
objectgridSessionMetadata 0      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 1      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 2      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 3      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 4      0      0      Primary      kr050578Node03Cell\kr050578Node05\containerServer01_C-2
서버 총계: 0 <0 B>

총 카탈로그 서비스 도메인 수: 0 <0 B>
<사용된 바이트 통계는 단순 오브젝트 또는 COPY_TO_BYTES 복사 모드를 사용 중인 경우에만 정확합니다.>

CWXS10040I: showMapSizes 명령이 완료되었습니다.
```

18. 여기까지 하시면 Adminagent 를 통해서 WXS 의 카탈로그 서버와 컨테이너 서버 설정 및 구동까지 문제없이 완료하신 것 입니다.

팁2) 일반적인 세션이나 Side Cache 같은 단순 시나리오 경우에는 복제본에서 읽기를 보통 수행하도록 설정하므로 원본과 복제 파티션이 각각 한쪽 컨테이너에 몰려있어도 상관없지만 In-Line Cache 시나리오와 같이 부하가 몰릴 수 있는 경우에는 하단과 같은 명령을 통해서 파티션의 밸런싱 조정이 가능합니다.

```
xscmd -c balanceShardTypes -cep kr050578:4811
```

```
C:\IBM\WebSphere85\XtremeScale\ObjectGrid\bin>xscmd -c balanceShardTypes -cep kr050578:4811
시작 시간 :2014-03-06 14:31:05.399

CWXS10068I: 명령 실행 중: balanceShardTypes

*** session 데이터 그리드 및 sessionMapSet 맵 세트에 대한 밸런싱 상태 명령의 결과 인쇄.
파티션 샤드 유형          이전 컨테이너
현재 컨테이너
-----
0      Primary              kr050578Node03Cell\kr050578Node05\containerServer01_C
-2 kr050578Node03Cell\kr050578Node05\containerServer02_C-2
0      AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer02_C
-2 kr050578Node03Cell\kr050578Node05\containerServer01_C-2
2      Primary              kr050578Node03Cell\kr050578Node05\containerServer01_C
-2 kr050578Node03Cell\kr050578Node05\containerServer02_C-2
2      AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer02_C
-2 kr050578Node03Cell\kr050578Node05\containerServer01_C-2

CWXS10040I: balanceShardTypes 명령이 완료되었습니다.
```

파티션 밸런싱 결과 확인

```
C:\IBM\WebSphere85\XtremeScale\ObjectGrid\bin>xscmd -c showMapSizes -cep kr050578:4811
시작 시간 :2014-03-06 14:31:35.731

CWXS10068I: 명령 실행 중: showMapSizes

*** session 데이터 그리드 및 sessionMapSet 맵 세트에 대한 결과 표시.

*** kr050578Node03Cell\kr050578Node05\containerServer02에 대한 맵 나열 ***
맵 이름          파티션 맵 항목 사용한 바이트 샤드 유형          컨테이너
-----
objectgridSessionMetadata 0          0          0          Primary              kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 1          0          0          AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 2          0          0          Primary              kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 3          0          0          AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer02_C-2
objectgridSessionMetadata 4          0          0          AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer02_C-2
서버 총계: 0 <0 B>

*** kr050578Node03Cell\kr050578Node05\containerServer01에 대한 맵 나열 ***
맵 이름          파티션 맵 항목 사용한 바이트 샤드 유형          컨테이너
-----
objectgridSessionMetadata 0          0          0          AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 1          0          0          Primary              kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 2          0          0          AsynchronousReplica  kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 3          0          0          Primary              kr050578Node03Cell\kr050578Node05\containerServer01_C-2
objectgridSessionMetadata 4          0          0          Primary              kr050578Node03Cell\kr050578Node05\containerServer01_C-2
서버 총계: 0 <0 B>

총 카탈로그 서비스 도메인 수: 0 <0 B>
<사용된 바이트 총계는 단순 오브젝트 또는 COPY_TO_BYTES 복사 모드를 사용 중인 경우에만 정확합니다.>

CWXS10040I: showMapSizes 명령이 완료되었습니다.
```

팁3) 해당 강좌는 관리의 편의를 위해서 IBM WAS Base 버전의 Adminagent 를 사용해서 WXS 카탈로그 서버와 컨테이너 서버의 설정/구동에 대해서 이야기 하고 있지만 당연히 이전 강좌에서 언급한 것처럼 하단과 같이 별도 스크립트를 이용해서도 설정/구동이 가능합니다.

WXS 카탈로그 서버 구동

```
startXSSErver catalogServer01 -catalogServiceEndPoints cs0:kr050578:6600:6601 -listenerPort 4811
```

WXS 컨테이너 서버 구동

```
startXSSErver    containerServer01    -catalogServiceEndPoints    kr050578:4811    -objectgridFile  
C:/IBM/WebSphere/eXtremeScale/ObjectGrid/session/samples/objectGrid.xml    -deploymentPolicyFile  
C:/IBM/WebSphere/eXtremeScale/ObjectGrid/session/samples/objectGridDeployment.xml    -jvmArgs    -  
Xms256m -Xmx512m -Xgcpolicy:gencon
```

```
startXSSErver    containerServer02    -catalogServiceEndPoints    kr050578:4811    -objectgridFile  
C:/IBM/WebSphere/eXtremeScale/ObjectGrid/session/samples/objectGrid.xml    -deploymentPolicyFile  
C:/IBM/WebSphere/eXtremeScale/ObjectGrid/session/samples/objectGridDeployment.xml    -jvmArgs    -  
Xms256m -Xmx512m -Xgcpolicy:gencon
```

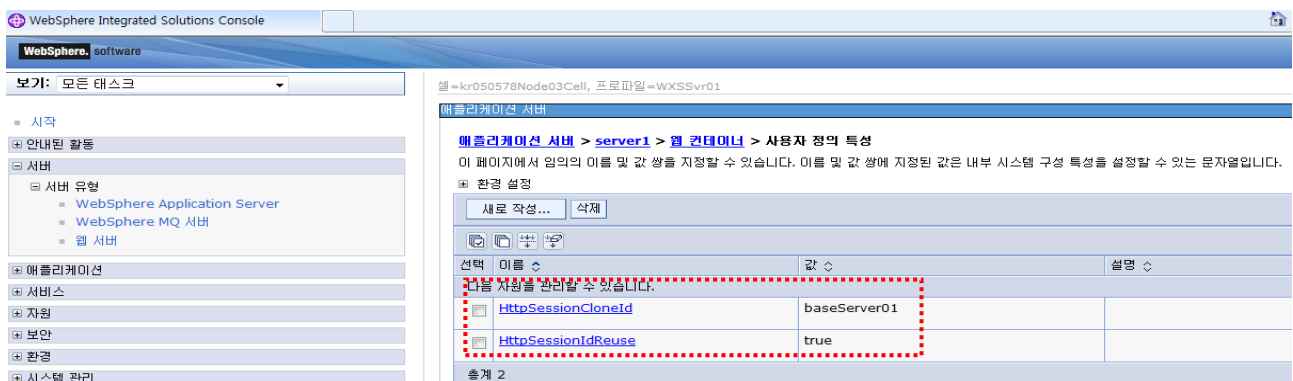
2) AdminAgent 를 이용한 IBM WAS & WXS 통합 관리 – Client 준비

1. WXS 의 카탈로그 서버와 컨테이너 서버에 대한 준비가 완료되었으므로 이제 실제 테스트를 해보기 위하여 client 역할을 하기 위해 미리 만들어둔 server1, server2 에 대한 설정 작업을 수행합니다.

2. IBM WAS base 서버를 가지고 Session Cluster 구축시에 각 WAS 서버 구분을 위해서 server1 과 server2 에 대해서 하단과 같은 옵션을 서버 > WebSphere Application Server > 해당 서버 > 웹 컨테이너 > 사용자 정의 특성 메뉴에 추가합니다.

HttpSessionCloneId = "구분 되는 이름" – Session Affinity 를 위한 구분되는 이름을 넣는 옵션

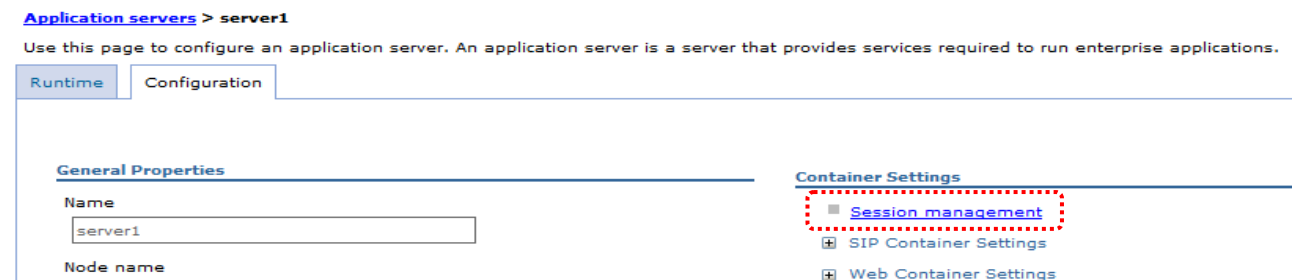
HttpSessionIdReuse = "true" – 두 대의 WAS 서버를 교차할 때도 SessionId 를 재사용하는 옵션



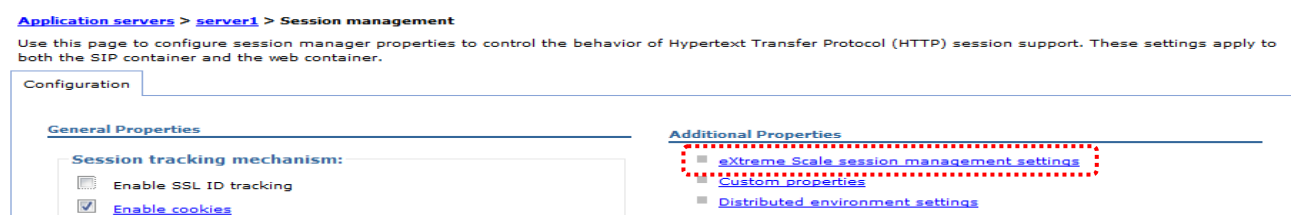
3. client 로 사용할 WAS 서버에 대해서 WXS 를 Session Management 용도로 사용하기 위한 설정을 추가 합니다.

(참고로 지금부터 설명될 부분을 하신다면 이후에 설명될 web.xml 수정 작업에서 objectGridType, catalogHostPort, objectGridName 설정을 삭제해도 정상 동작합니다. – 즉, 해당 부분은 어떻게 설정할 것이냐에 따라서 선택이 가능하다는 이야기 입니다.)

4. server1 의 서버 설정에 들어가서 Session management 메뉴를 선택합니다.



4. Session management 메뉴에서 eXtreme Scale session management setting 메뉴를 선택합니다.



5. Enable session management 를 선택하면 콤보박스를 선택할 수 있고 Remote 를 선택합니다.

Application servers > server1 > Session management > eXtreme Scale session management settings
Configure this server to be associated with eXtreme Scale.

Configuration

General Properties

☒ Enable session management

Manage session persistence by:

IBM WebSphere DataPower XC10 Appliance
 IBM WebSphere DataPower XC10 Appliance
 Remote eXtreme Scale data grid
 Embedded eXtreme Scale data grid

IBM WebSphere DataPower XC10 Appliance security credentials

* User name:
 * Password:
 Test Connection...

6. 이미 만들어진 WXS 카탈로그 서비스 도메인 이름을 선택하고 Browse 를 클릭합니다.

General Properties

☒ Enable session management

Manage session persistence by:
 Remote eXtreme Scale data grid

Remote eXtreme Scale data grid configuration

Configure this application to be associated with a remote eXtreme Scale data grid.

Catalog service domain that manages the remote session data grid:
 CatalogDomain01

Remote data grid in which to store session information:
 Browse...

7. 등록된 remote data grids 의 리스트가 나오는데 session 을 선택하고 설정을 완료합니다.

General Properties

☒ Enable session management

Manage session persistence by:
 Remote eXtreme Scale data grid

Remote eXtreme Scale data grid configuration

Configure this application to be associated with a remote eXtreme Scale data grid.

Catalog service domain that manages the remote session data grid:
 CatalogDomain01

Remote data grid in which to store session information:
 Browse...

List of active remote data grids

session

Close

8. 여기까지 설정하셨다면 client 로 사용될 WAS 에서 필요한 설정은 마무리 하신 것입니다. 이제 애플리케이션이 WXS 를 Session management 용도로 사용하기 위한 추가 작업이 이루어져야 합니다.

(추가 작업이 이루어진다고는 하지만 애플리케이션의 변경은 없으며 web.xml 의 변경만 필요합니다.)

9. 샘플 애플리케이션과 관련해서는 "[WAS8.5.5]하나씩 이해하는 IBM WAS v8.5.5 - 4. IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축 (http://www.websphere.pe.kr/xe/was_info_re/174609)" 강좌를 참고해서 애플리케이션의 web.xml 을 변경합니다. 또는, 하단의 snippet 을 애플리케이션의 web.xml 에 추가만하시면 WXS 를 이용하여 Session Clustering 이 가능합니다. 애플리케이션의 종류에 상관없이 추가가 가능하며 filter 를 쓰는 애플리케이션이 있을 경우 WXS 관련 filter 를 가장 먼저 타게 순서만 조정하시면 됩니다.

web.xml 추가 부분

```
<context-param>
    <param-name>sessionCreatedOnFailover</param-name>
    <param-value>>false</param-value>
</context-param>
<context-param>
    <param-name>securityEnabled</param-name>
    <param-value>>false</param-value>
</context-param>
<context-param>
    <param-name>fragmentedSession</param-name>
    <param-value>>true</param-value>
</context-param>
<context-param>
    <param-name>sessionTableSize</param-name>
    <param-value>10000</param-value>
</context-param>
<context-param>
    <param-name>objectGridType</param-name>
    <param-value>REMOTE</param-value>
</context-param>
<context-param>
    <param-name>replicationInterval</param-name>
    <param-value>10</param-value>
</context-param>
<context-param>
    <param-name>catalogHostPort</param-name>
```

```

    <param-value>kr050578:4811</param-value>
</context-param>
<context-param>
    <param-name>objectGridName</param-name>
    <param-value>session</param-value>
</context-param>
<filter>
    <description>Filter that provides for an ObjectGrid based Session Manager.</description>
    <display-name>HttpSessionFilter</display-name>
    <filter-name>HttpSessionFilter</filter-name>
    <filter-class>com.ibm.ws.xs.sessionmanager.HttpSessionFilter</filter-class>
</filter>
<filter-mapping>
    <filter-name>HttpSessionFilter</filter-name>
    <url-pattern>/*</url-pattern>
    <dispatcher>REQUEST</dispatcher>
</filter-mapping>
<listener>
    <description>IBMSessionListener</description>
    <display-name>IBMSessionListener</display-name>
    <listener-class>com.ibm.ws.xs.sessionmanager.IBMHttpSessionListener</listener-class>
</listener>

```

- 이미 언급한 것처럼 client 로 사용할 WAS 서버에서 WXS 를 Session management 로 쓰기 위한 설정을 하신 분들은 objectGridType, catalogHostPort, objectGridName 설정을 넣지 않아도 문제없이 동작합니다.

- **catalogHostPort** = kr050578:4811 (카탈로그 서비스가 수행되는 호스트 명과 port)

관리콘솔에서 카탈로그 서비스에 대한 설정을 이미 수행했다면 삭제해도 무방하며 만약 HA를 위해서 WXS 카탈로그 서버를 한대 이상 구성했다면 hostname:port, hostname:port 형식으로 입력하면 됩니다.

- **objectGridType** = REMOTE

REMOTE 나 EMBEDDED 를 선택할 수 있으며 REMOTE 인 경우 지금 서비스 하는 WAS 서버 외부에 session management 를 위한 WXS 컨테이너가 존재하는 것을 의미합니다. EMBEDDED 를 선택한 경우에는 지금 서비스 하는 WAS 서버 내부에 WXS 컨테이너가 존재하는 것을 의미합니다.

- **fragmentedSession** = true

true 인 경우는 session.setAttribute 가 호출될 때만 해당 attribute 가 복제되며 해당 값이 false 면 getSession 이 호출될 때마다 전체 session data 가 복제됩니다.

- **replicationInterval** = 10

Session 복제가 비동기식으로 10초 마다 수행됨. 값이 0인 경우 Servlet의 종료시점에 동기식으로 복제되기 때문에 성능저하 발생될 수 있습니다만 session 이 반드시 유지될 경우라면 0 설정도 가능합니다.

- **sessionTableSize** = 10000

WAS 에 설정된 session table size 최대값보다 큰 수 지정

10. 샘플 애플리케이션을 server1, server2 대상으로 배포시킨 후에 해당 서버를 구동시킵니다.

Select	Name ↕	Node ↕	Host Name ↕	Version ↕	Status ↕
You can administer the following resources:					
<input type="checkbox"/>	catalogServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	containerServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	containerServer02	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	server1	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	server2	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔

11. 테스트 서버가 시작되면 하단과 같이 session 을 사용하는 실제 샘플 애플리케이션을 호출하여 문제가 없는지 확인합니다.

<http://localhost:9081/MySessionTest/SessionTest.jsp>

← localhost:9081/MySessionTest/SessionTest.jsp	
Most Visited ⓘ 웹에서 PMR 확인 IBM IBM Information Cent... IBM WebSphere	
Your session details:	
Session Id:	1DsyHeVKwizur995o5ki6P2
Session creation date:	Thu Mar 06 14:55:09 KST 2014
Last time session accessed:	Thu Mar 06 14:55:09 KST 2014
Server Name	null
Session Count	1

12. 샘플 애플리케이션을 연속으로 호출하여 Session Count 가 증가하는 것을 확인합니다.



Your session details:

Session Id:	1DsyHeVKwjzur995o5ki6P2
Session creation date:	Thu Mar 06 14:55:09 KST 2014
Last time session accessed:	Thu Mar 06 14:55:31 KST 2014
Server Name	null
Session Count	9

13. xscmd 명령을 통해서 파티션의 구성과 Mapsize 를 확인해보면 하단과 같이 지금 생성된 Session 이 WXS 의 컨테이너 서버의 파티션에 실제 저장된 것을 확인할 수 있습니다.

xscmd -c showMapSizes -cep kr050578:4811

```

*** kr050578Node03CellWkr050578Node05WcontainerServer02 에 대한 맵 나열 ***
맵 이름          파티션 맵 항목 사용한 바이트 샤드 유형      컨테
이너
-----
objectgridSessionAttribute 2      1      312 B      Primary      kr05
0578Node03CellWkr050578Node05WcontainerServer02_C-2
objectgridSessionMetadata 0      0      0      Primary      kr05
0578Node03CellWkr050578Node05WcontainerServer02_C-2
objectgridSessionMetadata 1      0      0      AsynchronousReplica kr05
0578Node03CellWkr050578Node05WcontainerServer02_C-2
objectgridSessionMetadata 2      1      1 KB      Primary      kr05
0578Node03CellWkr050578Node05WcontainerServer02_C-2
objectgridSessionMetadata 3      0      0      AsynchronousReplica kr05
0578Node03CellWkr050578Node05WcontainerServer02_C-2
objectgridSessionMetadata 4      0      0      AsynchronousReplica kr05
0578Node03CellWkr050578Node05WcontainerServer02_C-2
서버 총계: 2 <2 KB>

*** kr050578Node03CellWkr050578Node05WcontainerServer01 에 대한 맵 나열 ***
맵 이름          파티션 맵 항목 사용한 바이트 샤드 유형      컨테
이너
-----
objectgridSessionAttribute 2      1      312 B      AsynchronousReplica kr05
0578Node03CellWkr050578Node05WcontainerServer01_C-2
objectgridSessionMetadata 0      0      0      AsynchronousReplica kr05
0578Node03CellWkr050578Node05WcontainerServer01_C-2
objectgridSessionMetadata 1      0      0      Primary      kr05
0578Node03CellWkr050578Node05WcontainerServer01_C-2
objectgridSessionMetadata 2      1      1 KB      AsynchronousReplica kr05
0578Node03CellWkr050578Node05WcontainerServer01_C-2
objectgridSessionMetadata 3      0      0      Primary      kr05
0578Node03CellWkr050578Node05WcontainerServer01_C-2
objectgridSessionMetadata 4      0      0      Primary      kr05
0578Node03CellWkr050578Node05WcontainerServer01_C-2
서버 총계: 2 <2 KB>

총 카탈로그 서비스 도메인 수: 4 <4 KB>

```

14. 해당 서버의 장애를 가정하고 server1 을 중지시킨 후에 미리 만들어둔 server2 의 HTTP 포트를 통해서 직접 샘플 애플리케이션을 호출합니다. WXS 를 활용하여 Session clustering 이 구성되어 있기 때문에 문제없이 Session count 가 증가되는 것을 확인할 수 있습니다. (테스트시에 IBM HTTP Server 를 사용하지 않았는데 해당 부분의 설정이 궁금하신 분들은 "[WAS8.5]하나씩 이해하는 IBM WAS v8.5.5 - 4. IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축" 강좌를 참고하시기 바라겠습니다.)

Select	Name	Node	Host Name	Version	Status
You can administer the following resources:					
<input type="checkbox"/>	catalogServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	containerServer01	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	containerServer02	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔
<input type="checkbox"/>	server1	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	✖
<input type="checkbox"/>	server2	kr050578Node05	kr050578	Base 8.5.5.1 WXS 8.6.0.4	➔

localhost:9084/MySessionTest/SessionTest.jsp

Most Visited 웹에서 PMR 확인 IBM IBM Information Cent... WebSphere

Your session details:

Session Id:	1DsyHeVKwjzur995o5ki6P2
Session creation date:	Thu Mar 06 14:55:09 KST 2014
Last time session accessed	Thu Mar 06 14:58:33 KST 2014
Server Name	null
Session Count	10

15. 뿐만 아니라 xscmd 명령을 활용하면 파티션에 실제 저장된 Session 의 값도 직접 조회 및 확인이 가능합니다.

xscmd -c findByKey -g session -m objectgridSessionAttribute -fs ".*" -rv -cep kr050578:4811

```
C:\W\IBM\WebSphere85\WextremeScale\WObjectGrid\bin>xscmd -c findByKey -g session -m
objectgridSessionAttribute -fs ".*" -rv -cep kr050578:4811
시작 시간 :2014-03-06 15:07:34.040

CWXSI0068I: 명령 실행 중: findByKey
1 일치 키를 찾았습니다.

파티션 키                                값
-----
2      1DsyHeVKwjzur995o5ki6P2/MySessionTest_COUNT 10

CWXSI0040I: findByKey 명령이 완료되었습니다.
```

16. 다시 한번 확인을 위해 샘플 애플리케이션을 호출해보고 xscmd 로 값을 확인해 봅니다.

localhost:9084/MySessionTest/SessionTest.jsp

Most Visited 웹에서 PMR 확인 IBM IBM Information Cent... WebSphere

Your session details:

Session Id:	1DsyHeVKwjzur995o5ki6P2
Session creation date:	Thu Mar 06 14:55:09 KST 2014
Last time session accessed	Thu Mar 06 15:09:24 KST 2014
Server Name	null
Session Count	11

```
C:\W\IBM\WebSphere85\WextremeScale\WObjectGrid\bin>xscmd -c findByKey -g session -m
objectgridSessionAttribute -fs ".*" -rv -cep kr050578:4811
시작 시간 :2014-03-06 15:09:43.806

CWXSI0068I: 명령 실행 중: findByKey
1 일치 키를 찾았습니다.

파티션 키                                값
-----
2      1DsyHeVKwjzur995o5ki6P2/MySessionTest_COUNT 11

CWXSI0040I: findByKey 명령이 완료되었습니다.
```

17. 여기까지 확인이 되셨다면 문제없이 AdminAgent 를 이용한 IBM WAS & WXS 의 Session Cluster 구축 및 통합 관리 파트를 잘마무리 하신 것 입니다.

3) 관리를 위한 모니터링

1. IBM WXS 의 경우에는 관리를 위한 전문 모니터링 도구를 같이 제공합니다.
2. IBM WXS 가 설치된 폴더 > ObjectGrid > bin 폴더에서 startConsoleServer.bat/sh 를 수행하여 모니터링을 위한 콘솔 서버를 시작시킵니다.

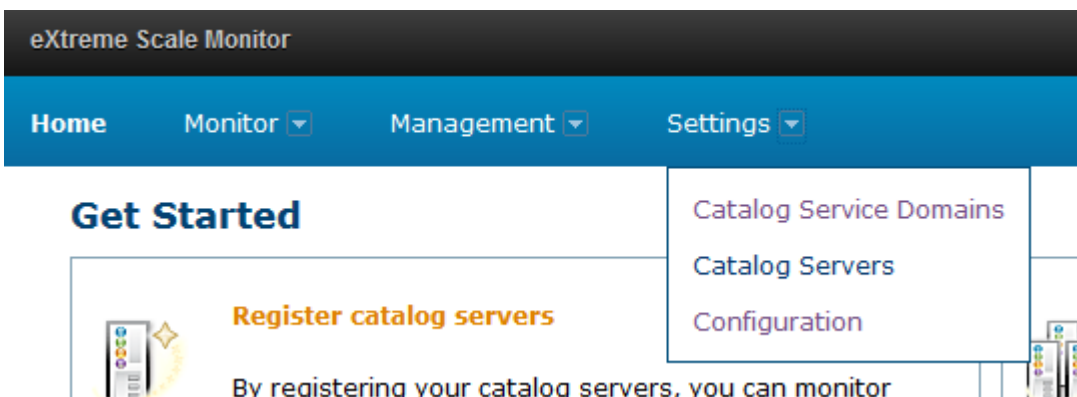
```
C:\IBM\WebSphere85\eXtremeScale\ObjectGrid\bin>startConsoleServer
XL TXE-J compiling stylesheet: file:///C:/IBM/WebSphere85/eXtremeScale/ObjectGrid/console/private/expanded/zero/zero.cli.tasks-1.1.1.6.31660/config/mg_info.xslt
CWPZT0600I: Command resolve was successful
C:\IBM\WebSphere85\eXtremeScale\ObjectGrid\bin>
```

3. WXS 모니터 콘솔에 접속합니다. <https://localhost:7443/login>

(기본 id 와 password 는 admin/admin 입니다.)



4. 초기 설정을 위해서 Settings > Catalog Servers 메뉴를 선택합니다.



5. 해당 메뉴가 나오면 왼쪽 상단에 있는 녹색 '+' 버튼을 클릭하여 카탈로그 서버를 새로 등록합니다.

eXtreme Scale Monitor

Home Monitor Management Settings

Catalog Servers +

Search... New

No results were found.

Registering an existing Catalog Server.

* Name: catalogServer01

* Host name: kr050578 ⓘ

* Listener port: 4811 ⓘ

OK Cancel

6. 카탈로그 서버 등록이 완료되면 Settings > Catalog Service Domains 메뉴를 클릭합니다.

eXtreme Scale Monitor

Home Monitor Management Settings

Catalog Service Domains +

Search... New

No results were found.

Registering a catalog service domain.

* Name: CatalogDomain01

OK Cancel

7. 이전과 마찬가지로 녹색 '+' 버튼을 클릭하여 카탈로그 서비스 도메인을 생성합니다.

eXtreme Scale Monitor

Home Monitor Management Settings

Catalog Service Domains +

Search... New

No results were found.

Registering a catalog service domain.

* Name: CatalogDomain01

OK Cancel

8. 생성된 카탈로그 서비스 도메인에 미리 만들어둔 카탈로그 서버를 입력하여 등록합니다.

eXtreme Scale Monitor

Home Monitor Management Settings

Catalog Service Domains

Search...

CatalogDomain01

CatalogDomain01

Name: CatalogDomain01

Catalog servers: (none)

Generator class: None provided

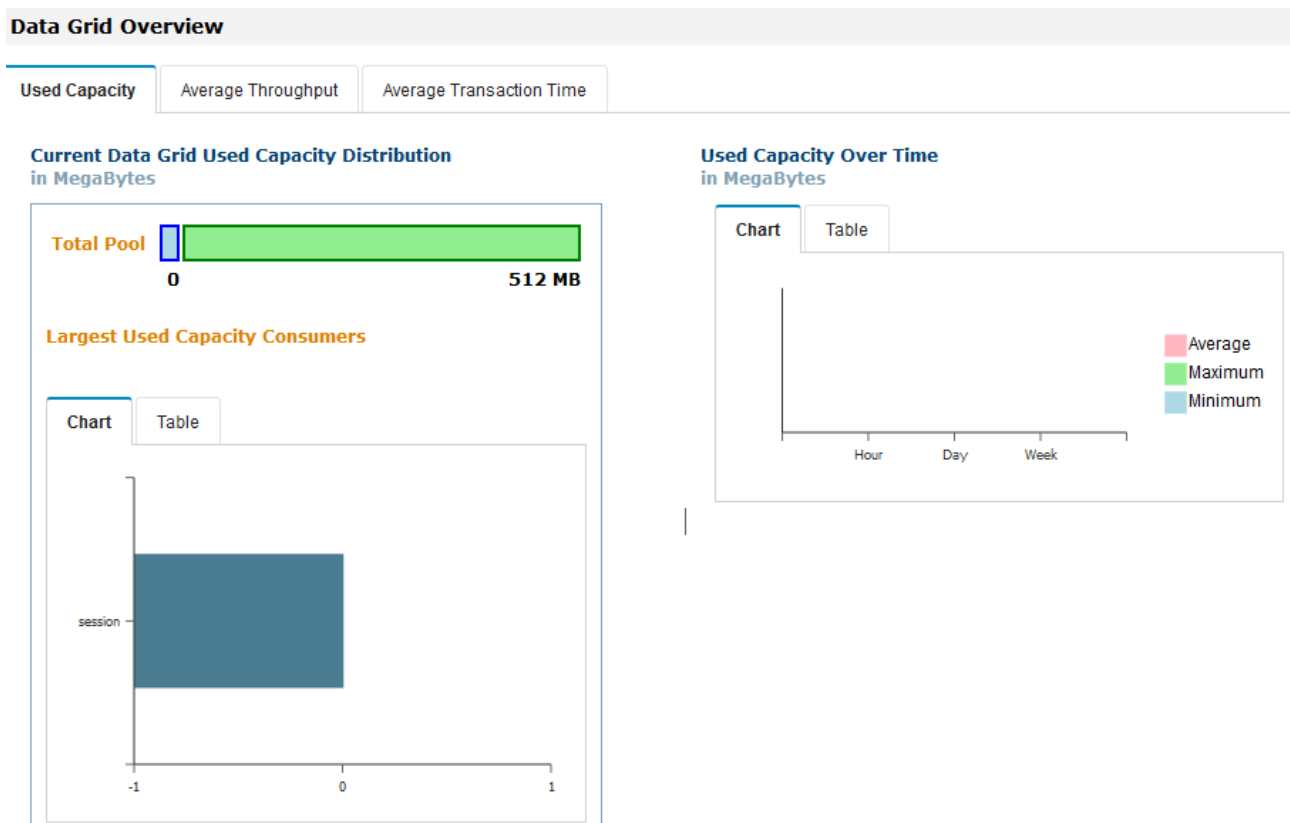
catalogServer01

9. 문제없이 등록이 완료되면 하단과 같이 우측 상단에 Connected 로 변경된 것을 확인할 수 있습니다.

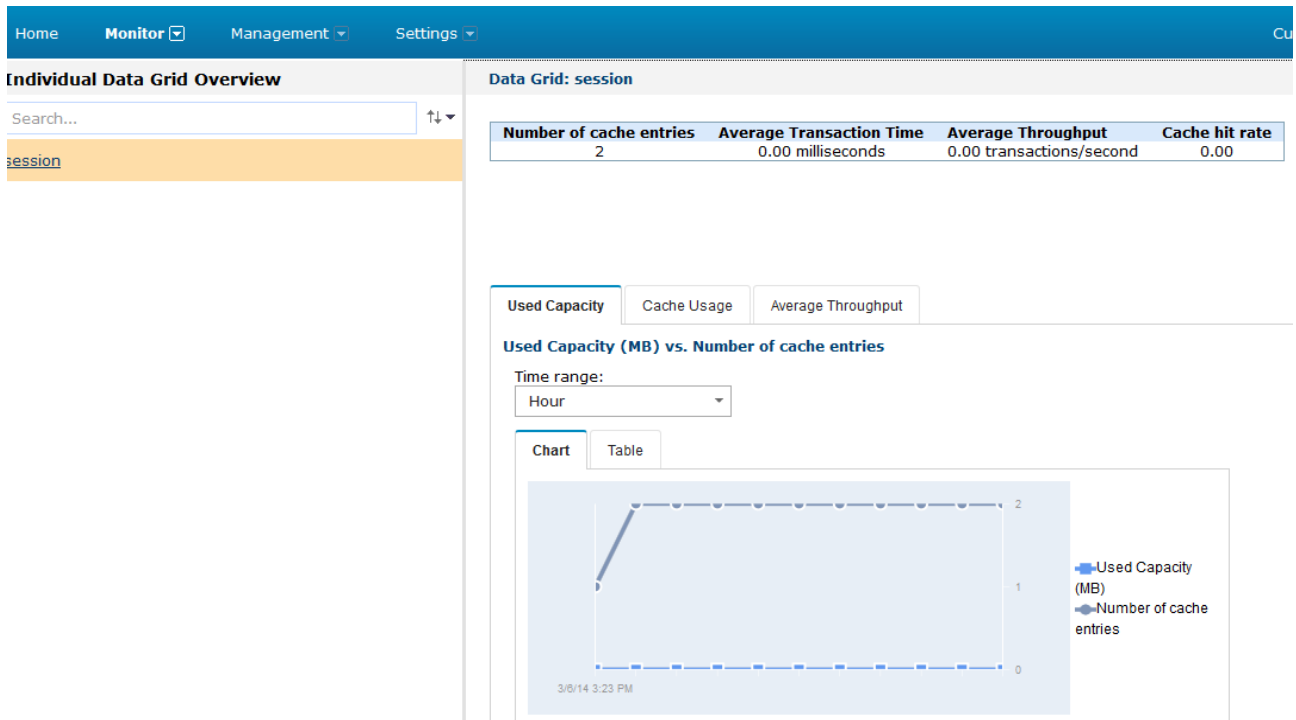
Welcome, Administrator Log Out IBM

Current catalog service domain CatalogDomain01 Connected

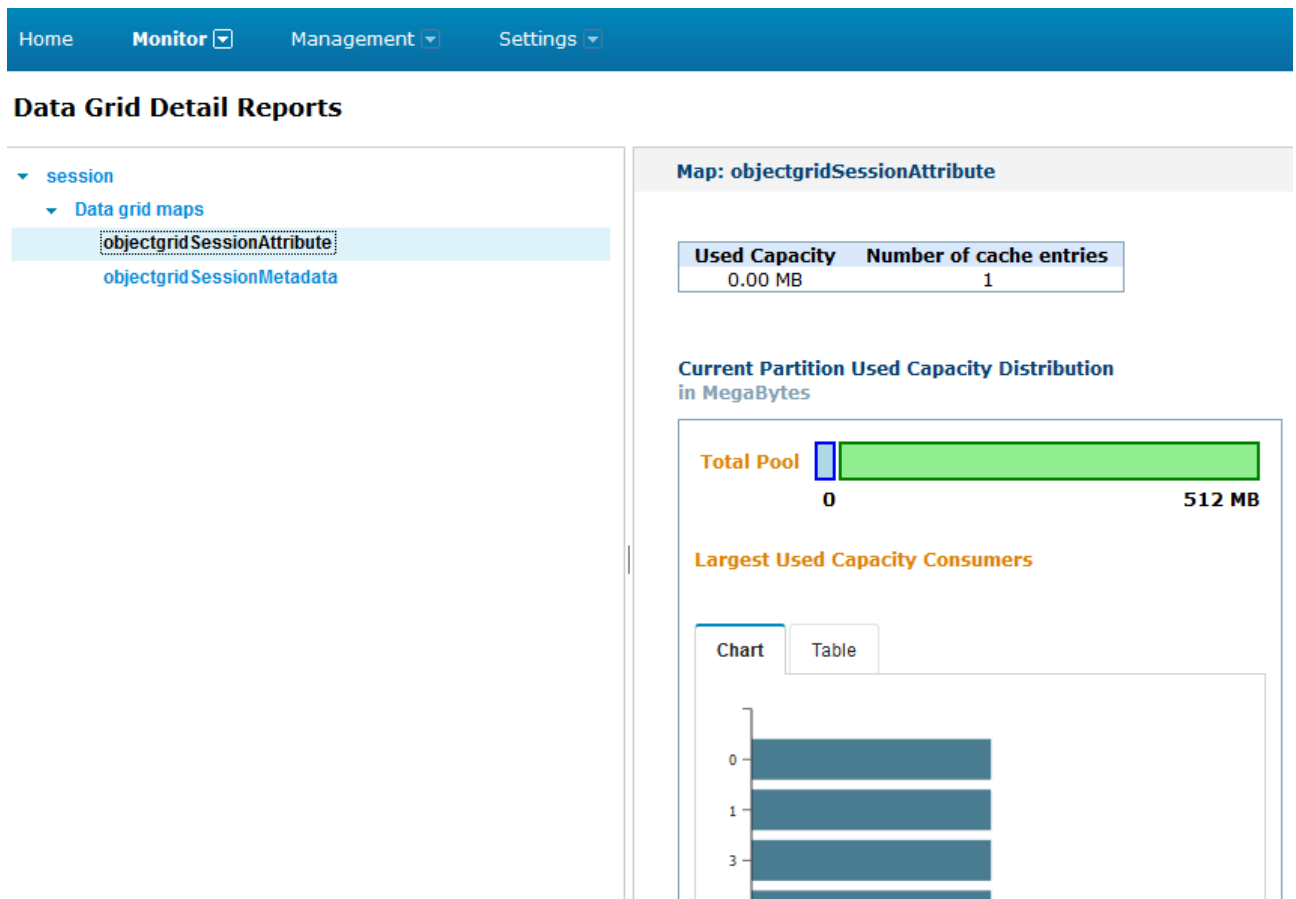
10. 등록이 완료되면 Monitor > Data Grid Domain Overview 를 클릭하여 전반적인 상황을 모니터링 할 수 있습니다.



11. Monitor > Data Grid Overview 메뉴를 통해서 만들어진 Data Grid 단위로 모니터링 할 수도 있습니다.



12. Monitor > Data Grid Details 메뉴를 통해서 만들어진 MapSet 단위로 모니터링 할 수도 있습니다.



13. Management > Query Data Grid Contents 메뉴를 통해서 실제 Data Grid 에 저장된 Cache 데이터에 대한 조회 및 확인이 가능합니다. 예를 들어 Data grid map 을 하나 선택한 후 "."* 를 검색하면 해당 map 에 있는 모든 데이터를 하단과 같이 확인할 수 있습니다.

Query Data Grid Contents

session

- Data grid maps
 - objectgridSessionAttribute
 - objectgridSessionMetadata

Map: objectgridSessionAttribute

Enter a regular expression to find keys in the map. After searching for keys, use the invalidate button to permanently remove entire query regardless of the number of matching keys.

.*

Regular expression help

☒ Invalidate ☐ Show values

Key	Partition
1DsyHeVKwjzur995o5ki6P2/MySessionTest_COUNT	2

14. 이때, 해당 Key 를 클릭하면 하단과 같이 실제 값도 같이 확인이 가능합니다.

Map: objectgridSessionAttribute

Enter a regular expression to find keys in the map. After searching for keys, use the invalidate button to permanently remove entire query regardless of the number of matching keys.

.*

Regular expression help

☒ Invalidate ☐ Show values

Entry Details

Key:
1DsyHeVKwjzur995o5ki6P2/MySessionTest_COUNT

Value:
11

15. Session 을 공유하지 않는 다른 브라우저에서 샘플 애플리케이션을 다시 호출하여 추가로 session 을 Data Grid 에 저장하게 되면 하단과 같이 바로 모니터링 콘솔에서 확인이 가능합니다.

Session Tester

localhost:9084/MySessionTest/SessionTest.jsp

Apps IEM developerWorks : W... PMR 확인 IEM WAS85 IEM WebSphere

Your session details:

Session Id:	A8qPGJcCSOkwVLw28m8pLHn
Session creation date:	Thu Mar 06 15:42:51 KST 2014
Last time session accessed:	Thu Mar 06 15:42:51 KST 2014
Server Name	null
Session Count	1

Map: objectgridSessionAttribute

Enter a regular expression to find keys in the map. After searching for keys, use the invalidate button to permanently remove entire query regardless of the number of matching keys.

.*

Regular expression help

☒ Invalidate ☐ Show values

Key	Partition
1DsyHeVKwjzur995o5ki6P2/MySessionTest_COUNT	2
A8qPGJcCSOkwVLw28m8pLHn/MySessionTest_COUNT	4

16. 마지막으로, 강좌를 진행하면서 이미 경험하셨겠지만 xscmd 명령어를 사용하시면 모니터링 콘솔이 제공하는 다양한 모니터링 기능을 cmd 화면에서도 즉각 확인이 가능합니다.

(다시 한번 강좌를 처음부터 훑어 보세요.^^&)

7) 결론

이번 강좌에서는 IBM WAS v8.5.5 에서 라이선스 적으로 통합된 WXS 를 이용해서 IBM WAS Base 버전과 WXS 를 이용한 Session Cluster 구축에 대하여 살펴보는 시간을 가져봤습니다. 이전까지 IBM WAS Base 버전을 이용해서 제한된 범위에서 Cluster 구축이 가능하긴 했지만 fail-over 만 지원했고 Session Persistence 는 DB 를 활용하는 경우만 지원하였기 때문에 많이 활용되는 경우가 드물었습니다.

그러나, IBM WAS v8.5.5 부터는 Base 버전이라고 해도 WXS 라이선스가 Session Cluster 용으로 통합 제공되므로 이번 강좌에서 맛 보신것과 같이 두 대 이상의 IBM WAS Base 버전을 사용한다고 해도 완벽한 Session Cluster 를 구축할 수 있습니다. 결국, 이를 통해 Base 버전이라고 해도 점점 더 다양한 용도와 안정성이 필요한 시스템에 사용될 수 있는 길이 열렸다고 생각합니다.

여기서는 WXS 의 Session 저장 목적 부분만 강조했지만 해당 제품은 맨 처음에 언급된 것처럼 In-Memory Data Grid 솔루션으로 해당 기능 말고도 다양한 기능들이 더 있습니다. 해당하는 좋은 기능은 향후 다른 강좌에서 살펴볼 수 있게 되길 바라며 이번 강좌는 마무리하고 다음 강좌에서 뵈도록 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 – WebSphere eXtreme Scale Version 8.6
(http://pic.dhe.ibm.com/infocenter/wxsinfo/v8r6/index.jsp?topic=%2Fcom.ibm.websphere.extreme.scale.doc%2Fwelcome%2Fwelcome_xs.html)
6. InfoCenter – HTTP session management
(<http://pic.dhe.ibm.com/infocenter/wxsinfo/v8r6/index.jsp?topic=%2Fcom.ibm.websphere.extreme.scale.doc%2Fcxshhttpsession.html>)