

하나씩 쉽게 따라 해보는 IBM WebSphere Application Server(WAS) v7 – 16

이정운 (juwlee@kr.ibm.com)

하나씩 쉽게 따라 해보는 IBM WAS v7 그 열 여섯번째 이야기를 시작합니다. 열 여섯번째 이야기는 지난 강좌에서 Server clustering 만 구축되고 생략된 Session clustering 구축입니다. 이전 강좌에서도 이야기 했지만 IBM WAS ND v7 으로 Server clustering 을 구축하게 되면 서버를 병렬로 구성하여 성능도 높일 수 있으며 서버 장애에 대한 대처도 할 수 있습니다. 다만, 여기서 한 가지 생각해 보실 게 있는데 바로 어플리케이션에서 Session 을 사용할 경우입니다. Server clustering 만을 구축하였다면 병렬로 위치한 다른 서버에 의해서 서비스는 지속할 수 있지만 Session 을 보존하고 다른 서버로 넘겨줄 수 있는 방법이 없기 때문에 장애가 발생되면 Session 을 소실할 수 밖에 없습니다.

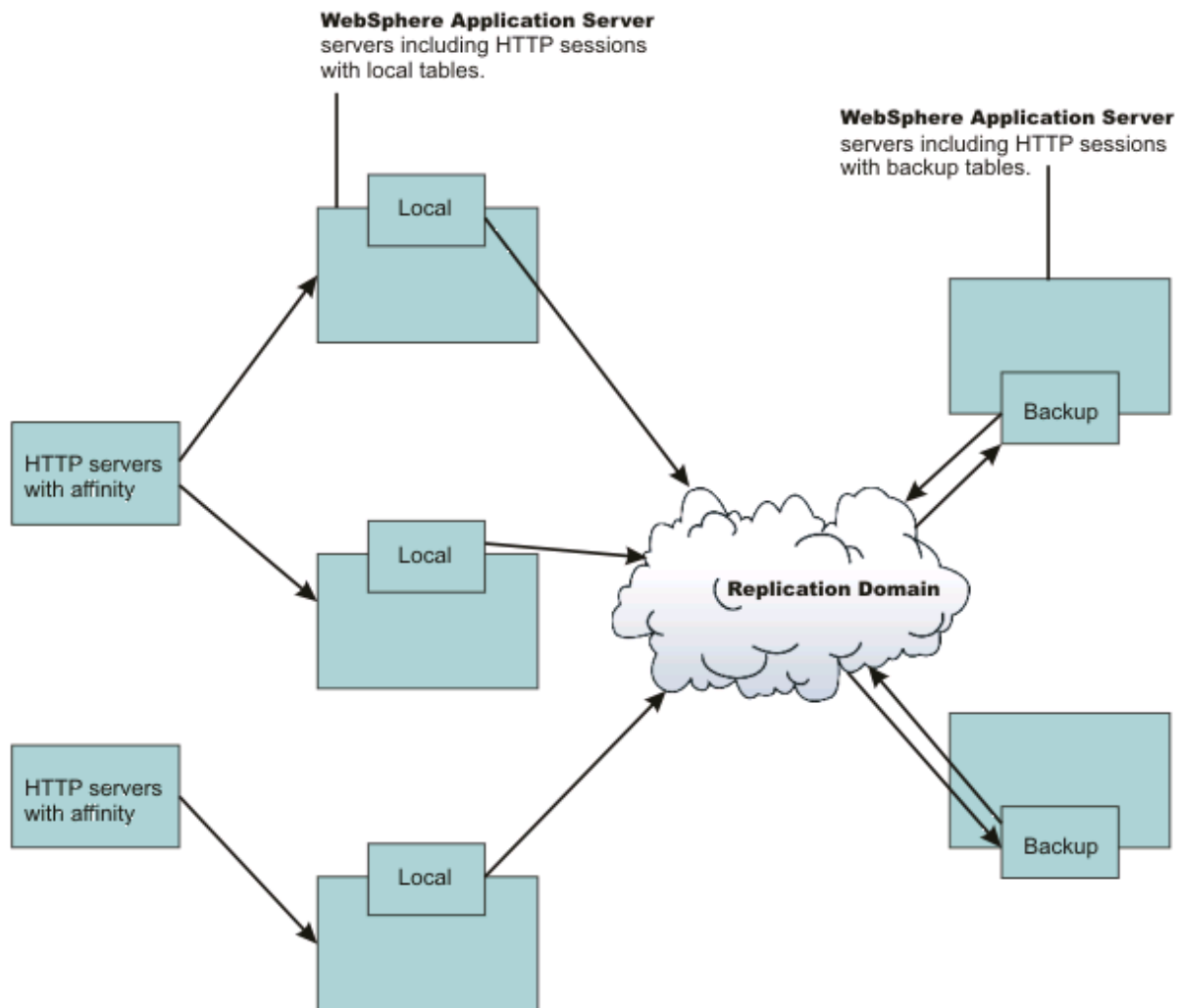
여기서 좀 더 이해하기 편하도록 잠시 Session 이야기를 하고 넘어가도록 하겠습니다. Session 은 이미 잘 알고 계시겠지만 WAS 서버 단에서 클라이언트의 연결 정보나 상태를 가지고 있는 객체(Object) 입니다. 인터넷 쇼핑몰 등에서 자주 볼 수 있는 장바구니를 생각하시면 이해하시기가 편하신데, 한번 로그인을 하면 장바구니에 상품을 넣어두었다가 브라우저를 닫고 다시 접속해도 장바구니 페이지를 보시면 장바구니가 비어있는 것이 아니라 계속적으로 장바구니에 내가 쇼핑한 상품들을 확인할 수 있습니다. 이것이 바로 Session 을 이용한 것입니다. 즉, Session 은 클라이언트가 접속을 하면 클라이언트 정보를 서버에 Session 객체에 저장을 해두고 동일한 클라이언트가 접속을 하면 해당 Session 을 계속 사용할 수 있게 해서 Stateless(상태가 저장되지 않음) 라는 HTTP 프로토콜의 한계성을 극복하고 계속적으로 여러 요청에 대하여 클라이언트가 마치 자신의 상태를 보존하는 것처럼 보여주는 기술입니다.

위에서도 말씀드렸지만, 어플리케이션에 대한 Session 은 서버에 생성되기 때문에 서버가 장애가 나면 클라이언트의 상태를 저장해 둔 Session 을 잃어버리게 됩니다. 그 말은 결국, 장바구니에 넣어놓은 상품을 다 잃어버리는 일이 발생할 수 있다는 것이죠. 장바구니의 경우는 어느정도 괜찮지만 이게 만약 돈에 관련된 정보라면 피해가 클수 있겠죠. 이를 막기 위해서 WAS 에서는 Session Clustering 기능을 제공합니다. Session Clustering 기능이란 Session 을 해당 서버에만 보존하는 것이 아니라 이웃한 다른 서버나 Session 서버, DB에 저장하여 장애가 발생하면 저장된 위치에서 Session 을 복구하여 다른 서버로 처리가 되는 경우라도 동일한 Session 을 유지하여 장바구니에 넣어놓은 상품을 그대로 다 볼 수 있는 기능입니다. 이렇게 되는 경우에만 실질적으로, 고객이 뒀 단의 WAS 장애가 나든, 안나든 모른채 그대로 정확하게 서비스를 유지할 수 있습니다. (장애가 발생할 때 가장 중요한 것은 실제 사용자인 고객이 장애를 인지하지 못하게 서비스를 계속 유지하는 것입니다.)

IBM WAS ND v7 은 Session Clustering 을 위하여 크게 Memory-to-Memory 방식과 DB Centric 방식을 다 지원합니다. Memory-to-Memory 방식은 다시 2가지 방식으로 나누어지는데 peer-to-peer 방식과 Client Server 방식입니다. Peer-to-peer 방식은 말 그대로 지정된 모든 서버가 Session 을 각 서버마다 다 보관하는 방식이며 Client Server 방식은 Server 로 지정된 서버가

Session 서버가 되어서 Client 로 지정된 서버의 Session 을 별도로 보관하는 것입니다. 이 때, Server 로 지정된 Session Server 는 서비스를 하는 것이 아니라 Session 을 보관하는 전용 서버로 사용되는 것입니다. 마지막으로 DB Centric 은 Session 을 절대 유실되지 않아야 하는 경우에 많이 사용되는 것으로 Session 이 생성될 때마다 DB에다가 Session 을 복사해 두고 장애가 발생하면 DB 에서 Session 을 가지고 와서 복원하는 방식입니다. 이렇듯 각 방법 마다 장, 단점이 있는 것이고 고객사의 환경에 따라 필요한 방식으로 골라서 쓰시면 됩니다.

이번 강좌에서 테스트로 보여드릴 방식은 Client Server 방식입니다. 서비스를 하는 서버(Client로 지정) 에서 Session 이 생성될 때마다 Session 서버(Server로 지정)에 저장을 하는 방식으로, 서버에 장애가 발생하면 Session 서버에 저장해 놓은 Session을 가용한 다른 서버로 보내서 그 쪽에서 계속적인 클라이언트 요청을 중단 없이 처리하는 방식입니다. 이 방법은 안정성도 높고 성능도 좋아서 기업에서 많이 사용되는 방법중에 하나입니다. 하단의 간략한 구조 그림을 참조하시면 좀 더 이해가 편하실 것 입니다.



참조:

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/topic/com.ibm.websphere.zseries.doc/info/zseries/ae/cprs_m2m_cs.html

Part 1. Session Clustering 설정

Session Clustering 을 설정하시려면 관리콘솔에서 작업을 하셔야 합니다. 따라서 먼저 Dmgr 을 시작시킵니다. (시작 방법은 이전 강좌를 참조하세요.) 관리 콘솔이 시작되면 Servers > WebSphere Application Servers 메뉴를 클릭하여 하단과 같이 서버들이 등록된 것을 확인합니다. 여기서 중요한 점이 있는데 이전에 자동 생성된 server1 을 Session 을 보관할 Session 서버로 사용할 예정이라는 것 입니다. 만약, server1 이 없거나 지우신 경우라면 New 를 클릭하여 새로운 서버를 만드시고 해당 서버를 Session Server 로 사용하면 됩니다.

Session Clustering 을 설정하기 위해서 testServer01 로 되어있는 서버명을 클릭하여 서버 세부 설정 메뉴로 들어갑니다.

Integrated Solutions Console Welcome aa

Cell=T400B92090901Cell01, Profile=Dmgr01

Application servers

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

Select	Name	Node	Host Name	Version	Cluster Name	Status
<input type="checkbox"/>	server1	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7		
<input type="checkbox"/>	testServer01	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	
<input type="checkbox"/>	testServer02	T400B92090901Node02	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	

하단과 같이 서버 세부 설정 메뉴가 나오면 Session management 를 클릭합니다.

[Application servers](#) > testServer01

Use this page to configure an application server. An application server is a server that provides service enterprise applications.

Configuration

General Properties

Name

testServer01

Node name

T400B92090901Node01

☒ Run in development mode

Container Settings

☒ Session management

☐ SIP Container Settings

☐ Web Container Settings

☐ Portlet Container Settings

☐ EJB Container Settings

Session management 세부 메뉴가 나오면 Session Clustering 을 설정하기 위하여 Distributed environment settings 를 선택합니다. Session 을 사용하시려면 쿠키가 사용가능 상태여야 하기 때문에 왼편에 enable cookies 가 선택된 것을 확인하고 넘어갑니다.

[Application servers](#) > [testServer01](#) > **Session management**

Use this page to configure session manager properties to control the behavior of Hypertext Transfer Protocol support. These settings apply to both the SIP container and the Web container.

Configuration

General Properties

Session tracking mechanism:

☐ Enable SSL ID tracking

☒ [Enable cookies](#)

☐ Enable URL rewriting

☐ Enable protocol switch rewriting

Maximum in-memory session count:
 sessions

☒ Allow overflow

Session timeout:

☐ No timeout

☒ Set timeout

minutes

Additional Properties

☐ [Custom properties](#)

☒ [Distributed environment settings](#)

Distributed environment settings 세부 메뉴를 확인하면 여러 옵션들이 보이는데 Client Server 방식은 결국, memory-to-memory replication(메모리 대 메모리 복제) 방식이기 때문에 복제 도메인 (Replication domain) 을 우선 설정하기 위하여 해당 부분을 클릭합니다.

[Application servers](#) > [testServer01](#) > [Session management](#) > **Distributed environment settings**

Use this page to specify how session data is saved in a distributed environment. The SIP container uses only memory-to-memory replication for distributed sessions.

Configuration

General Properties

Distributed sessions

☒ None

☐ [Database](#) (Supported for Web container only.)

☐ [Memory-to-memory replication](#) (No replication domains have been defined.)

Additional Properties

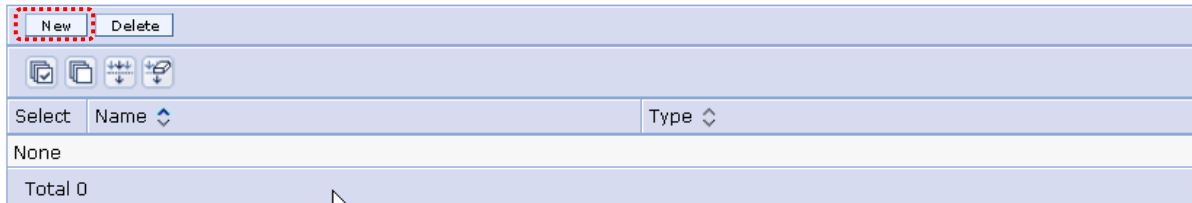
☒ [Custom tuning parameters](#)

보통의 경우 하단과 같이 등록된 복제 도메인이 없으므로 새롭게 추가하기 위하여 New 버튼을 클릭합니다.

[Application servers](#) > [testServer01](#) > [Session management](#) > [Distributed environment settings](#) > [Replication domains](#)

Use this page to create a list of replication domains that the session manager uses for replication of the dynamic cache service, and the stateful session bean failover components. All components that need to share information must be in the same replication domain.

Preferences

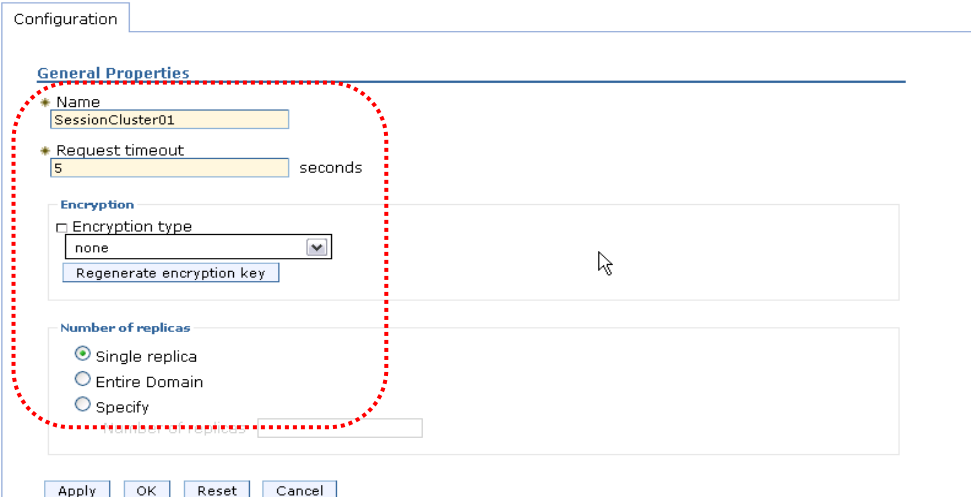


The screenshot shows a table for managing replication domains. At the top, there are 'New' and 'Delete' buttons. Below them are icons for adding, deleting, and refreshing. The table has columns for 'Select', 'Name', and 'Type'. Currently, the table is empty, showing 'None' in the Name column and 'Total 0' at the bottom.

New 버튼을 클릭하여 나온 설정 부분에 Name 을 넣고 Single replica 로 되어 있는 것을 확인하고 OK 를 눌러서 설정을 저장합니다.

[Application servers](#) > [testServer01](#) > [Session management](#) > [Distributed environment settings](#) > [Replication domains](#) > [New](#)

Use this page to configure the replication properties that are used by all of the components of this replication domain.

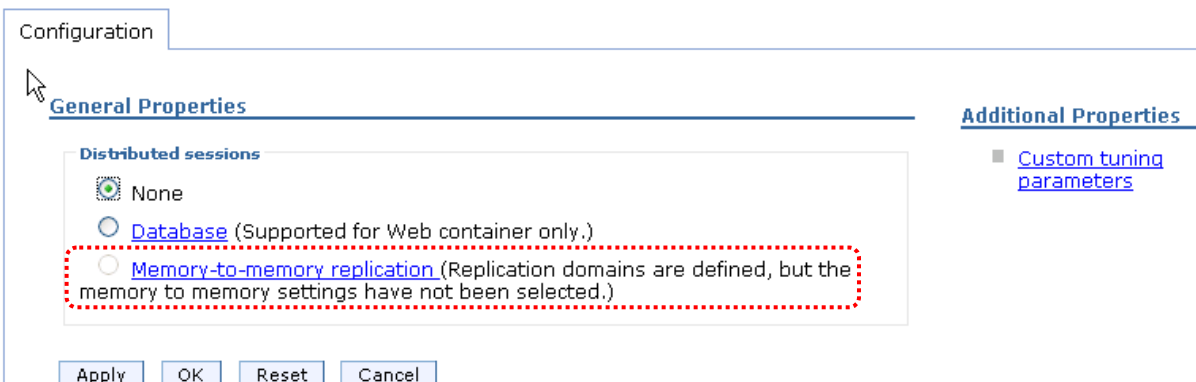


The screenshot shows the 'New' configuration dialog for a replication domain. It has a 'Configuration' tab. Under 'General Properties', the 'Name' field is set to 'SessionCluster01' and the 'Request timeout' is set to '5' seconds. The 'Encryption' section shows 'Encryption type' set to 'none' and a 'Regenerate encryption key' button. The 'Number of replicas' section has 'Single replica' selected. At the bottom are 'Apply', 'OK', 'Reset', and 'Cancel' buttons.

설정을 저장하고 다시 이전 메뉴로 돌아오면 하단처럼 memory-to-memory replication 명령이 변경된 것을 확인할 수 있습니다. (이전에는 복제 도메인이 없다는 것이고 지금은 지정되었다라는 의미입니다.) 다시 memory-to-memory replication 을 클릭합니다.

[Application servers](#) > [testServer01](#) > [Session management](#) > [Distributed environment settings](#)

Use this page to specify how session data is saved in a distributed environment. The SIP container uses only memory-to-memory replication for distributed sessions.



The screenshot shows the 'Distributed environment settings' configuration page. It has a 'Configuration' tab. Under 'General Properties', the 'Distributed sessions' section has three radio buttons: 'None', 'Database (Supported for Web container only.)', and 'Memory-to-memory replication (Replication domains are defined, but the memory to memory settings have not been selected.)'. The 'Memory-to-memory replication' option is selected and highlighted with a red dashed box. At the bottom are 'Apply', 'OK', 'Reset', and 'Cancel' buttons. On the right, there is a link for 'Additional Properties' > [Custom tuning parameters](#).

그럼 memory-to-memory replication 설정 메뉴가 이제서야 나오는데 만들어 놓은 복제 도메인을 선택하고 복제 모드를 Client only 라고 합니다. (Client only 로 설정하는 것은 Session 을 복제하는 클라이언트 역할 만을 하며, Session 복제본을 가지고 있는 Server 가 아니라 Session 을 만들어서 Session Server 에 보내는 서버라는 의미입니다.)

[Application servers](#) > [testServer01](#) > [Session management](#) > [Distributed environment settings](#) > **Memory-to-memory replication**

Use this page to configure memory-to-memory replication for failure recovery.

Configuration

General Properties

* Replication domain
SessionCluster01

* Replication mode
Client only

Apply OK Reset Cancel

위와 동일하게 testCluster 에 포함된 다른 한 서버인 testServer02 도 testServer01 와 똑같이 설정합니다.

[Application servers](#) > [testServer02](#) > [Session management](#) > [Distributed environment settings](#) > **Memory-to-memory replication**

Use this page to configure memory-to-memory replication for failure recovery.

Configuration

General Properties

* Replication domain
SessionCluster01

* Replication mode
Client only

Apply OK Reset Cancel

Client Server 모드로 Session Clustering 을 구성할 것이기 때문에 Session 서버로 활용할 server1 서버는 Server only 모드로 설정합니다. (Server only 모드의 의미는 Session 을 생성하거나 복제하지는 않고 Client only 모드로 지정된 서버의 Session 복제본을 보관하는 Session 서버로 활용한다는 의미입니다.)

[Application servers](#) > [server1](#) > [Session management](#) > [Distributed environment settings](#) > **Memory-to-memory replication**

Use this page to configure memory-to-memory replication for failure recovery.

Configuration

General Properties

* Replication domain
SessionCluster01

* Replication mode
Server only

Apply OK Reset Cancel

이렇게 하시면 Client Server 모드의 Session Clustering 설정을 모두 완료하신 것 입니다.

참고적으로 테스트를 통해서 바로 바로 Session clustering 이 되는 것을 확인하기 위하여 간단하게 튜닝 옵션을 조정합니다.

[Application servers](#) > [testServer01](#) > [Session management](#) > [Distributed environment settings](#) > [Tuning parameters](#) > [Custom tuning parameters](#)

Use this page to specify tuning parameters for session management.

Configuration

General Properties

Write frequency

☒ End of servlet service

☐ Manual update

☐ Time based:

10seconds

Write contents

☐ Only updated attributes

☒ All session attributes

Schedule sessions cleanup:

☐ Specifies distributed sessions cleanup schedule

First time of day (0-23):

Second time of day (0-23):

Apply

OK

Reset

Cancel

Write frequency 는 Session 이 Session 복제 서버에 저장되는 주기를 의미하며 기본으로 Time based 로 되어있어서 주기적으로 저장됩니다. 이를 end of servlet service 로 선택하여 Servlet 서비스가 끝나면 바로 생성된 Session 을 Session 복제 서버에 저장하도록 합니다.

Write contents 는 어떤 정보들을 저장할 것인가를 설정하는 것입니다. Update 된 것들만, 즉 변경이 있는 Session 만 저장할 것인지 그냥 모든 Session 을 저장할 것인지를 설정하는 것입니다.

위의 옵션들을 조정하면 테스트시에 바로 Session clustering 의 효과를 확인하실 수 있습니다.

Part 2. Session Clustering 을 위한 어플리케이션 설치 및 테스트

Session Clustering 을 위한 어플리케이션은 Session 을 사용하는 어떠한 어플리케이션이여도 상관없습니다. 여기서 좀 더 재미있고 Active 한 강좌를 진행해 보도록 하겠습니다. 참여지향 강좌를 한번 해보도록 하죠. 제가 이번 강좌에서는 Application 을 공개하지 않도록 하겠습니다. 다만, 쉽게 만드실 수 있도록 필요한 몇가지 방법들만 알려드리도록 하겠습니다. Session Clustering 테스트를 위한 어플리케이션은 한번 직접 제작해 보시기 바라겠습니다. (잘 모르겠다고 생각하시는 분들은 이전에 올린 개발 관련 강좌를 참조하시면 조금 더 도움이 될 수 있습니다.)

Session 을 만드는 방법 (Servlet 에서)

```
HttpSession httpSession = request.getSession();
```

WAS Server 이름을 받아올 수 있는 방법 (Servlet에서)

```
getServletContext().getAttribute("com.ibm.websphere.servlet.application.host")
```

Session ID 를 받아올 수 있는 방법 (Servlet 에서)

```
httpSession.getId()
```

어플리케이션을 설치하는 방식은 이전 강좌에서 설명한 방법과 동일하며 다만, Server mapping 부분만 하단처럼 testCluster 와 webserver1 을 선택해 주면 됩니다. 이렇게 하는 의미는 Session Clustering 때문이기 보다는 IBM WAS ND 를 이용해 Server Clustering 을 구축 했고 앞 단에 Web Server 를 위치시켜서 라우트 하는 Achitecture 이기 때문에 앞 단에 등록된 Web Server 에도 어플리케이션이 설치된 것을 알리기 위해서 맵핑 하는 것입니다.

(위의 단계를 진행하면 이전 강좌에서 이야기 한 것처럼 관리콘솔을 통해서 plugin-cfg.xml 을 생성 및 전파해주고 IHS 를 재시작해야 합니다. 이렇게 하는 이유는 plugin-cfg.xml 에 실질적으로 요청이 들어오면 어떤 서버로 보내주어야 하는가라는 정보가 있기 때문에 그부분의 update 를 위해서 입니다.)

Specify options for installing enterprise applications and modules.

Step 1 Select installation options

Step 2: Map modules to servers

Step 3 Summary

Map modules to servers

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=T400B92090901Cell01,cluster=testCluster
- WebSphere:cell=T400B92090901Cell01,node=T400B92090901Node01,server=server1
- WebSphere:cell=T400B92090901Cell01,node=T400B92090901Node01,server=webserver1

Apply

Select	Module	URI	Server
<input type="checkbox"/>	TyibBmtWeb	TyibBmtWeb.war,WEB-INF/web.xml	WebSphere:cell=T400B92090901Cell01,node=T400B92090901Node01,server=webserver1 WebSphere:cell=T400B92090901Cell01,cluster=testCluster

Previous Next Cancel

어플리케이션 설치를 완료하면 server1 부터 시작해서 테스트를 위하여 모든 서버를 시작 시킵니다. (당연히 IHS 도 시작되어 있어야 합니다.)

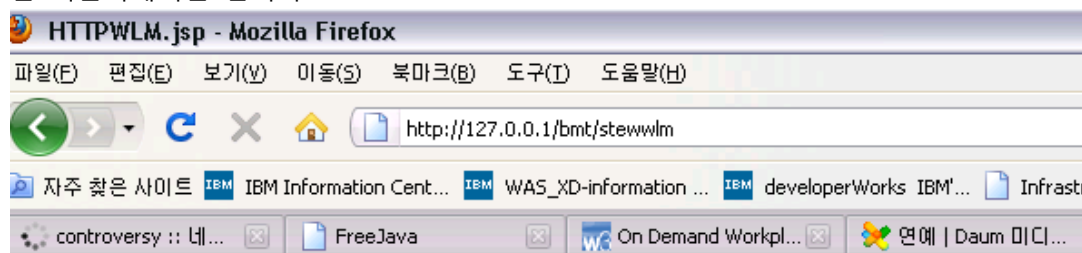
Application servers

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

<div> <div>NewDeleteTemplates...StartStopRestartImmediateStopTerminate</div> <div> </div> </div>						
Select	Name	Node	Host Name	Version	Cluster Name	Status
You can administer the following resources:						
<input type="checkbox"/>	server1	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7		
<input type="checkbox"/>	testServer01	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	
<input type="checkbox"/>	testServer02	T400B92090901Node02	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	
Total 3						

브라우저를 이용해서 Session 을 이용하는 어플리케이션을 호출 합니다. 지금 제가 사용하는 어플리케이션은 Session 을 사용하는 간단한 어플리케이션으로 호출 되는 Server 명과 횟수를 출력하는 어플리케이션 입니다.



V5 STEW BRAINDEAD HTTP WLM SERVLET

stewwlm

Select a Method of Execution:

- ☐ Display Server Name Only
- ☒ Display Server Name & Establish Session Affinity
- ☐ Invalidate Session ID and Affinity

Server Name = testServer01

No Count in Session

No Session requested

파일(F) 편집(E) 보기(V) 이동(S) 북마크(B) 도구(T) 도움말(H)

← → ↺ × 🏠 📄 http://127.0.0.1/bmt/stewwlm?source=SES

자주 찾은 사이트 IBM IBM Information Cent... IBM WAS_XD-information ... IBM dev


네이버 :: 세상의 ... FreeJava 할후 일상생활을 ...

Select a Method of Execution:

- ☐ Display Server Name Only
- ☒ Display Server Name & Establish Session Affinity
- ☐ Invalidate Session ID and Affinity

Session ID = FjrVoKT-gVDdZrRXO9KG7Ql

Messages

 An error occurred while stopping T400B92090901Node01/testServer01. Check the error logs for more information. [View JVM logs](#) for further details.
try terminate instead

⊕ Preferences

New

Delete

Templates...

Start

Stop

Restart

ImmediateStop

Terminate

Select	Name	Node	Host Name	Version	Cluster Name	Status
You can administer the following resources:						
<input type="checkbox"/>	server1	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7		
<input type="checkbox"/>	testServer01	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	
<input type="checkbox"/>	testServer02	T400B92090901Node02	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	

Total 3

HTTPWLM.jsp - Mozilla Firefox

파일(F) 편집(E) 보기(V) 이동(S) 북마크(B) 도구(T) 도움말(H)

http://127.0.0.1/bmt/stewwlm?source=SES

자주 찾은 사이트 IBM IBM Information Cent... IBM WAS_XD-information ... IBM dev...

네이버 :: 세상의 ... FreeJava 향후 일상생활을 ...

Select a Method of Execution:

- ☐ Display Server Name Only
- ☒ Display Server Name & Establish Session Affinity
- ☐ Invalidate Session ID and Affinity

Session ID = FjrVoKT-gVDdZrRXO9KG7Ql

그럼, 다시 testServer01 을 살리게 되면 어떻게 될까요? 하단처럼 testServer01 을 다시 시작 시킵니다.

⊕ Preferences

New

Delete

Templates...

Start

Stop

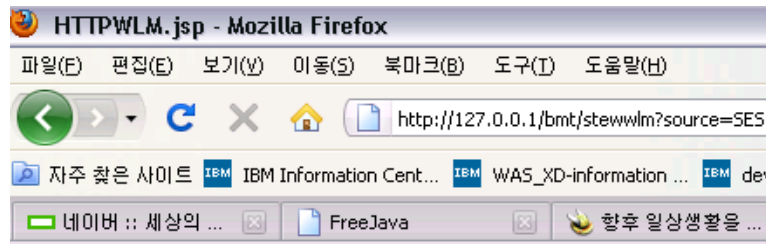
Restart

ImmediateStop

Terminate

Select	Name	Node	Host Name	Version	Cluster Name	Status
You can administer the following resources:						
<input type="checkbox"/>	server1	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7		
<input type="checkbox"/>	testServer01	T400B92090901Node01	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	
<input type="checkbox"/>	testServer02	T400B92090901Node02	T400B92090901.kr.ibm.com	ND 7.0.0.7	testCluster	
Total 3						

Server Name 이 다시 testServer01 로 변경된 것과 Count 는 그대로 증가하고 Session ID 도 동일하다는 것을 확인할 수 있습니다.



V5 STEW BRAINDEAD HTTP WL



Select a Method of Execution:

- ☐ Display Server Name Only
- ☒ Display Server Name & Establish Session Affinity
- ☐ Invalidate Session ID and Affinity

Server Name = testServer01

Count: 7 (from existing session)

Session ID = FjrVoKT-gVDdZrRXO9KG7QI

지금 보시는 것 처럼 장애가 발생한 서버의 Session 이 서비스가 가능한 다른 서버에서 사용되다가 다시 장애가 복구되어 정상적으로 서비스가 가능한 상태가 되면 Session 도 복원되는데, 이것을 보통 Fail-back 이라고 합니다.

여기까지 잘 따라오셨다면 Session Clustering 설정 및 테스트를 무사히 완료하신 것 입니다. 장애가 발생하여 Session 이 Fail-over 되는 것도 확인하셨고 다시 해당 서버가 복구 된 후에 Session 이 Fail-back 되는 것도 직접 확인하셨습니다. 이전에 열 세번째 강의부터 이어온 Server Clustering 이 열 다섯번째 강좌에서 겨우 마무리가 되었다는 말을 하였지만, 실질적으로 Session 을 쓰는 경우의 Server Clustering 이라면 Server Clustering 과 Session Clustering 을 같이 포함한 의미이기 때문에 진정한 마무리는 이번 강좌에서 이루어진 것 입니다. 그럼 이번 강좌는 여기서 마무리 하겠습니다. 이만~~~~~ ^^&

참고 1) IBM WebSphere Application Server v7.0 InfoCenter

http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?topic=/com.ibm.websphere.nd.multipiplatform.doc/info/welcome_nd.html

참고 2) IBM WebSphere Application Server v7.0 InfoCenter

-> Memory-to-memory topology: Client/server function

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/topic/com.ibm.websphere.zseries.doc/info/zseries/ae/cprs_m2m_cs.html

※이 자료의 저작권은 작성자에게 있으며 유포는 자유로이 허용되나 상업적으로 이용은 금합니다.