2008 WebSphere Services Technical Conference
world class skill building and technical enablement
May 5 - 9, 2008 • Las Vegas, NV

WebSphere V7

WSC Impressions of WAS V7

Thanks to
Don Bagwell
IBM Americas Advanced Technical Support
Washington Systems Center
dbagwell@us.ibm.com

WebSphere software

IBM

Agenda of Session

- Quick Overview of What's New in V7.0
- Review of Cell Planning and Construction
- Closer Look at "zDiff" Items WSC is Testing
- Wrap-Up with Miscellaneous Topics
- Questions and Answers / Discussion

Overview of What's New

And What Remains the Same

What Remains the Same

Let's set a baseline with some essential things that remain the same in V7.0:

- **Same essential server structure -- DMGR, Node Agent, AppServer, Daemon**
- **Same topology considerations -- servers, nodes, clusters and cells**
- **Same essential address space structure -- controller, servant, adjunct**
- **Same essential method of construction -- zPMT and batch job submission**
- **Same essential data connection mechanisms -- JDBC, JCA, JMS***

We're not looking at massive architectural shift. Most existing skills easily transferable.

And within that framework there's much new with V7.

Let's do a quick survey of what some of those things are.

*** There is a new MQ mechanism -- Resource Adapters**

Transition …

New Spec Levels in V7.0

V7 brings to the table many of the new evolving standards:

Specification	*V6.1*	*V7.0*
IBM Java SDK	SDK 5	SDK 6
Java Platform, Enterprise Edition (Java EE) specification	J2EE 1.4	Java EE 5
Java Platform, Standard Edition (Java SE) specification	J2SE 5	J2SE 6
Java Servlet specification	2.4	2.5
EJB	2.1	3.0
Portlet specification	1.0	2.0
JDBC	3.0	4.0
Java API for XML Web Services (JAX-WS) specification	2.0	2.1
Web Services Atomic Transaction (WS-AT)	1.0	1.1
Web Services Business Activity (WS-BA)	1.0	1.1
Web Services Coordination (WS-COOR)	1.0	1.1
Web Services for Java Platform, Enterprise Edition (JSR 109)	1.1	1.2
Web Services Policy Namespace	n/a	1.5
Web Services Addressing - Metadata	n/a	1.0
Web Services Atomic Transaction Version	n/a	1.0 and 1.1
Web Services Reliable Messaging Policy Assertion Version	n/a	1.1
WS-SecurityPolicy	n/a	1.2
WS-MakeConnection Version 1.0	n/a	1.0
JavaBeans Activation Framework (JAF)	1.0.2	1.1
OASIS WS-SecureConversation	n/a	1.3
OASIS WS-Trust	n/a	1.3
Java Naming and Directory Interface (JNDI) Specification	SE 5	SE 6
Java Transaction API (JTA) specification	1.0.1	1.1

Note: where the spec level in V7.0 is the same as V6.1, the specification is not shown on this chart.

Transition …

Java 6

IBM Java SDK 6 for z/OS brings a couple of things of noteworthy interest:

Shared Class Caching

Ability to share class objects in common cache accessible by multiple JVMs. Has the potential to significantly reduce the region size of servers if high percentage of class objects are common between servers.

A good deal of WebSphere infrastructure class objects can be shared cached.

Ahead-of-Time (AOT) class compiling

JITed code stored away. Can reduce the startup time of servers.

AOT code can be stored in shared class cache.

More Information

`http://publib.boulder.ibm.com/infocenter/javasdk/v6r0/index.jsp`

Transition …

"zDiff" Line Items

Several z/OS-only things have been placed into the V7 product:

New SMF 120 Subtype 9 Record
Better information collected and lower overhead

Thread Hang Recovery
A way to avoid abending servant region when a thread hangs

FRCA Caching in Controller
Exploitation of TCP FRCA caching from WebSphere z/OS Controller

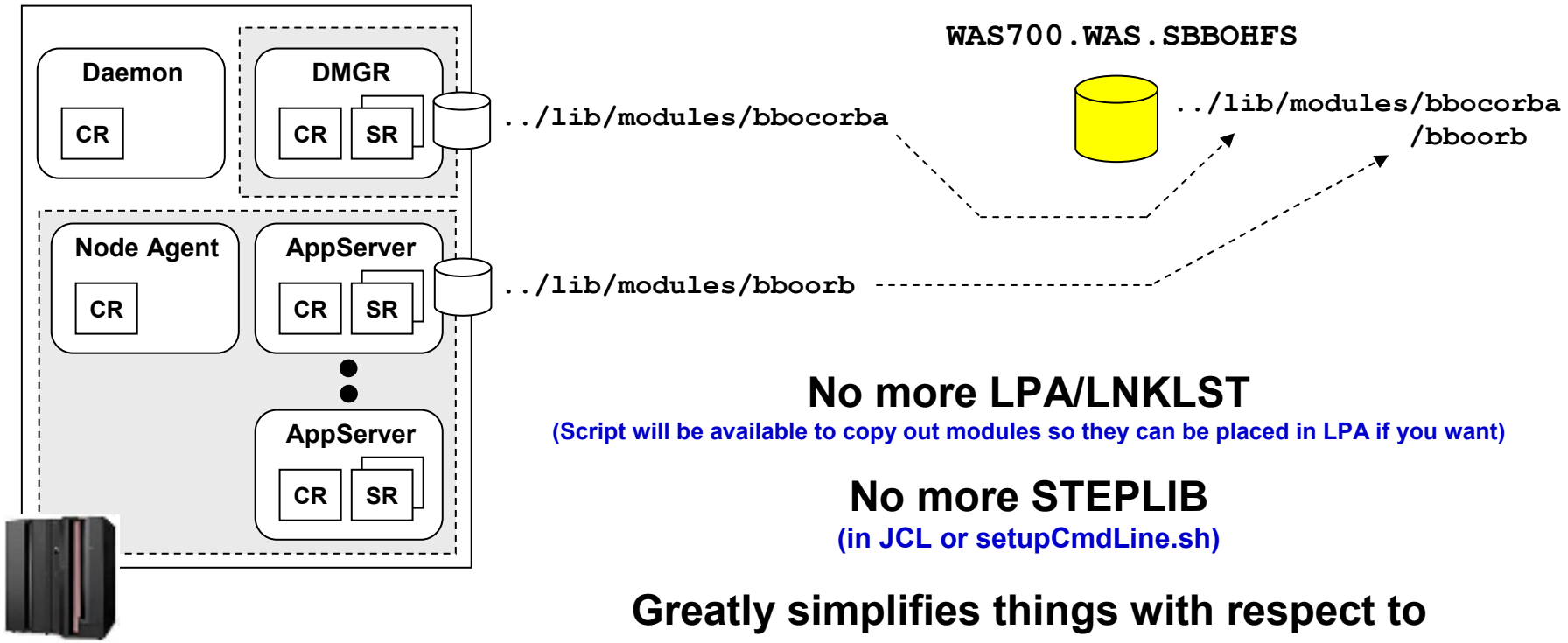
DCS signalling over XCF
Lower-overhead method of HAM signalling -- less burdensome large topologies

Separate section on this later

Transition …

No More Load Module Libraries

The load modules for V7 are now included in the HFS under `/lib/modules`.



No more LPA/LNKLST

(Script will be available to copy out modules so they can be placed in LPA if you want)

No more STEPLIB

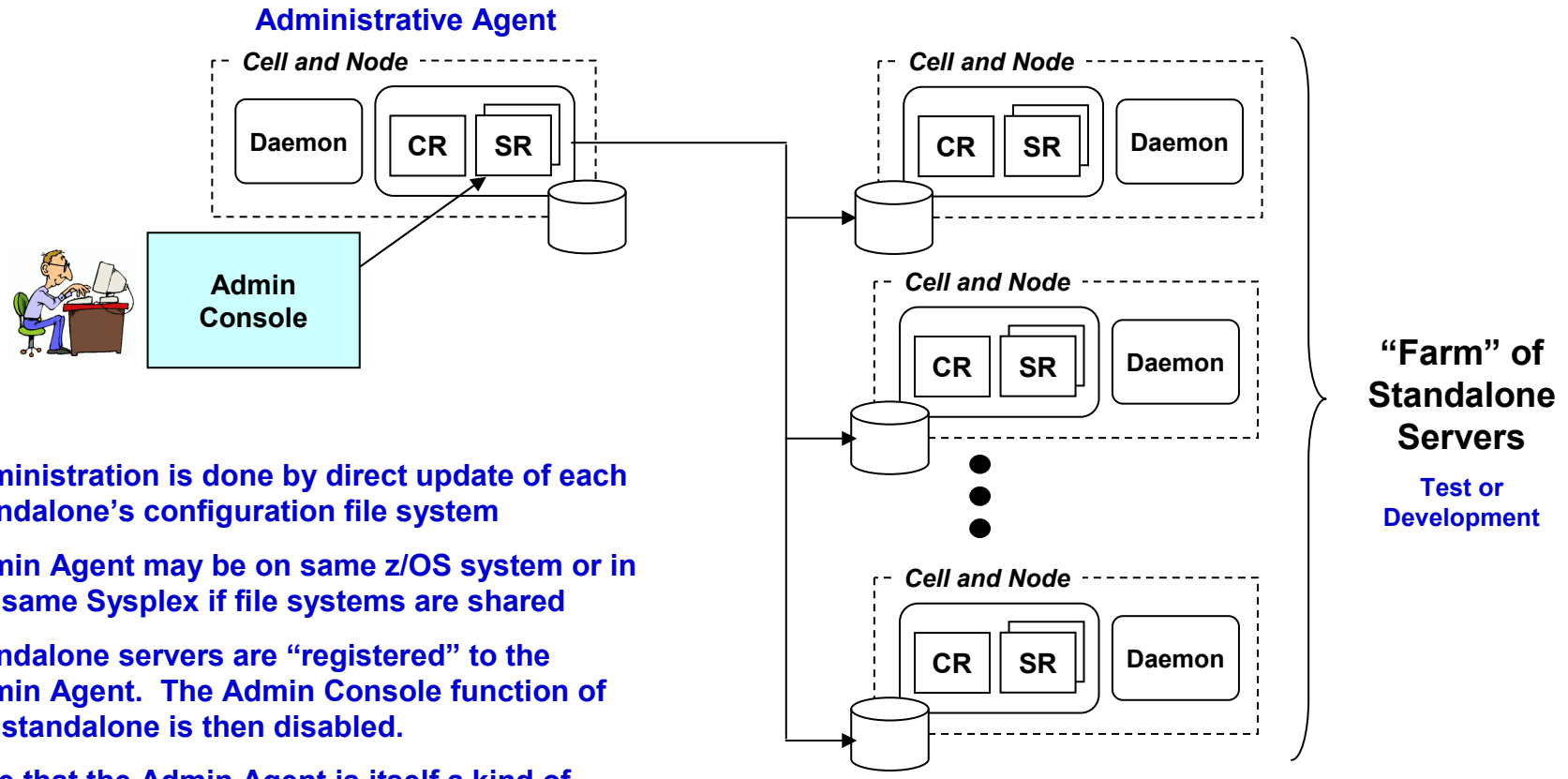
(in JCL or setupCmdLine.sh)

Greatly simplifies things with respect to rolling in maintenance

Transition …

"Flexible Management" - Administrative Agent

Think of the Administrative Agent as an Admin Console that can be switched to manage many different Standalone Server environments:

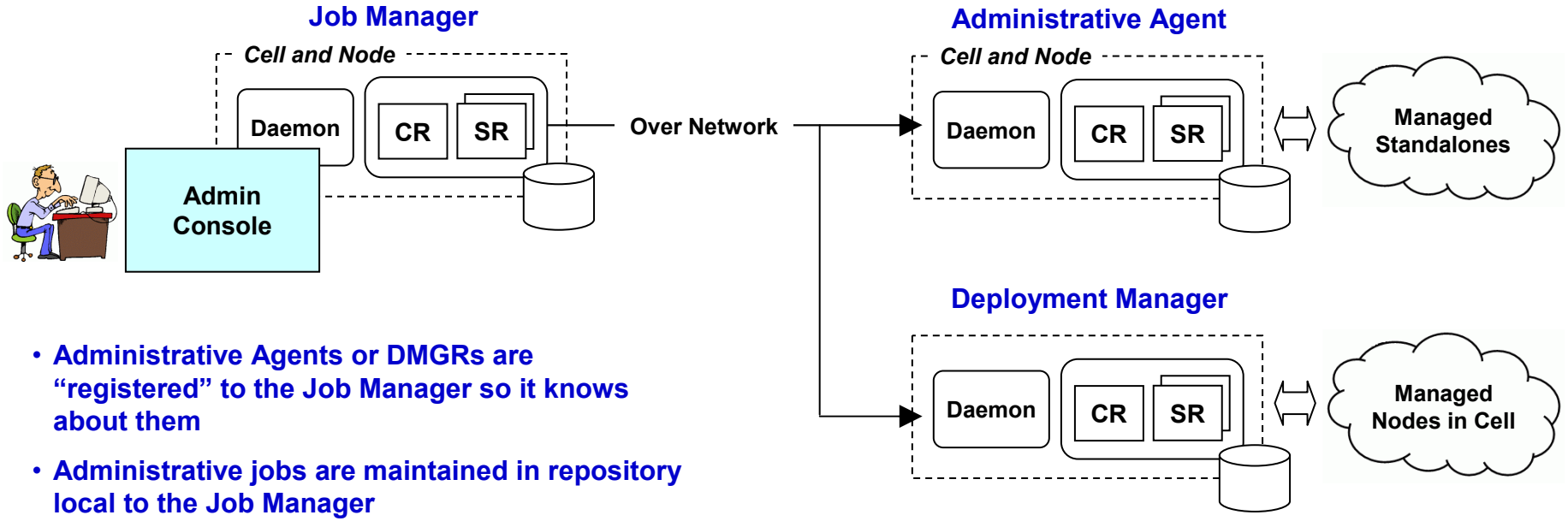


- Administration is done by direct update of each Standalone's configuration file system
- Admin Agent may be on same z/OS system or in the same Sysplex if file systems are shared
- Standalone servers are "registered" to the Admin Agent. The Admin Console function of the standalone is then disabled.
- Note that the Admin Agent is itself a kind of Standalone Server … its own cell/node/server structure with a Daemon

Transition …

"Flexible Management" - Job Manager

The idea is to have a central place to administer the asynchronous submission of WSADMIN "jobs" to multiple locations in your WebSphere environment:



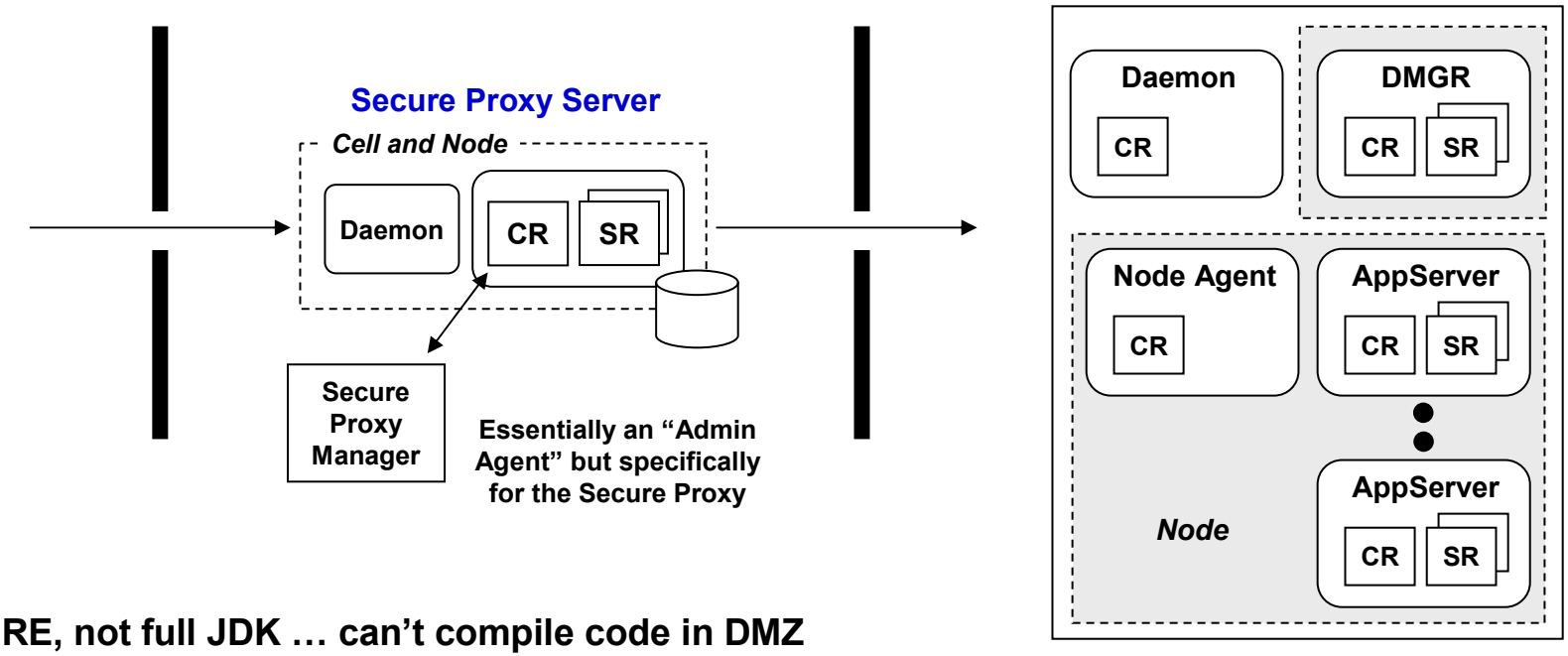
- **Administrative Agents or DMGRs are "registered" to the Job Manager so it knows about them**
- **Administrative jobs are maintained in repository local to the Job Manager**
- **You may then "submit the jobs" asynchronously to the targets and:**
  - **Set the job submission to take effect at a specified time.**
  - **Set the job submission to expire at a specified time.**
  - **Have the job submission occur at a specified time interval.**
  - **Notify the administrator through e-mail that the job has completed.**

Again, note that the Job Manager is its own cell/node/server structure. Naming and port planning applies here as well.

Transition …

Secure Proxy and its Manager

This is designed to be a more "DMZ Friendly" WebSphere Proxy device:



- JRE, not full JDK … can't compile code in DMZ
- No web container
- Not administered from Admin Console … use separate Admin Agent in DMZ (or local WSADMIN)
- Switches to unpriviledged user after binding to low-order ports
- Static routing (based on XML) or dynamic (requires DCS port kept open)

Transition …

Cell Planning and Construction

Basic Message

In many ways the planning and construction of a V.70 cell is similar to a V6.1 cell

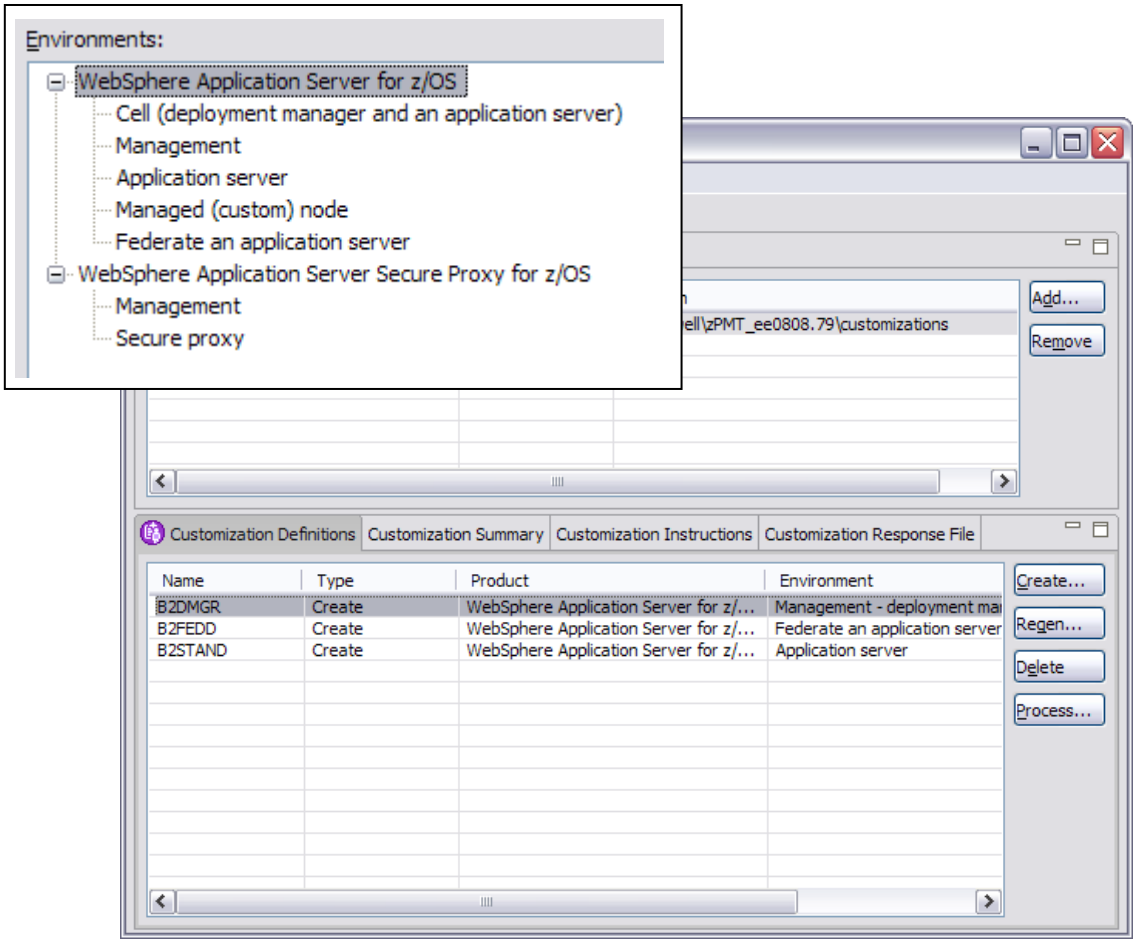
A few differences exist:

- **ISPF panels are now officially removed**
- **zPMT packaging different … though essential concept the same … a few interesting things to note**
- **Generated jobs similar but with a few noteworthy differences**
- **JCL start procedures slightly different**

Transition …

Profile Management Tool

The ISPF panels are gone … the Profile Management Tool now does all the customization work



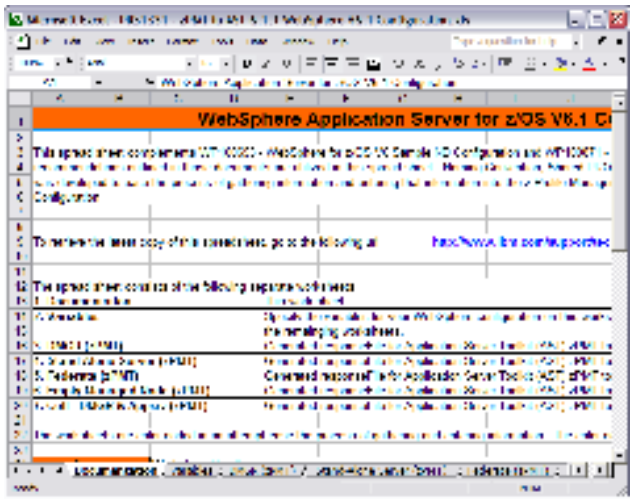
- Similar in look and feel to the V6.1 zPMT
- Now called WCT - WebSphere Customization Tools
- No longer part of the AST … that means a smaller footprint and better responsiveness
- New customizations or migration of existing
- V7.0 as well as V6.1
- Uploaded jobs are similar to before, with a few differences

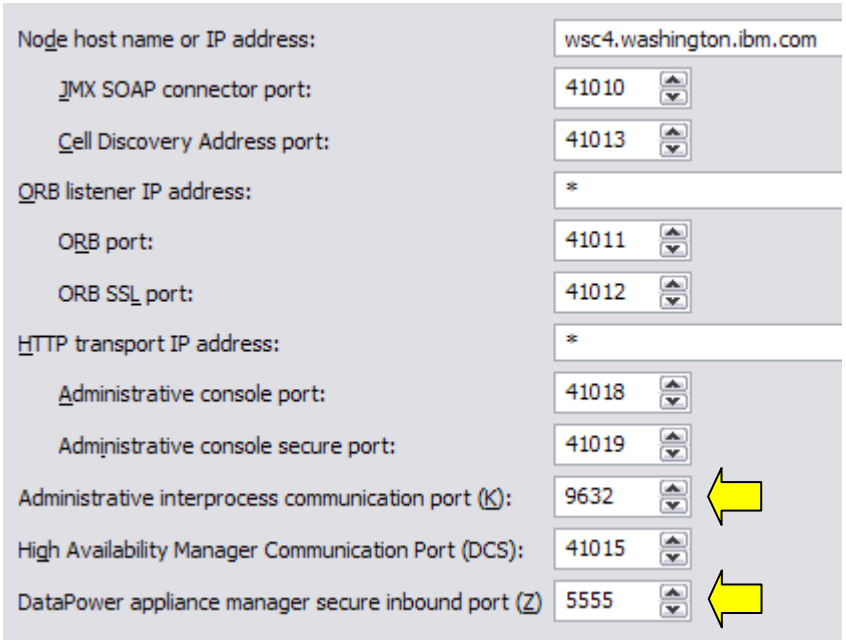
Transition …

The Planning Spreadsheet

The V6.1 edition works well for the standard cell construction. You do have to manually update a very few ports.

For example, a DMGR configuration:





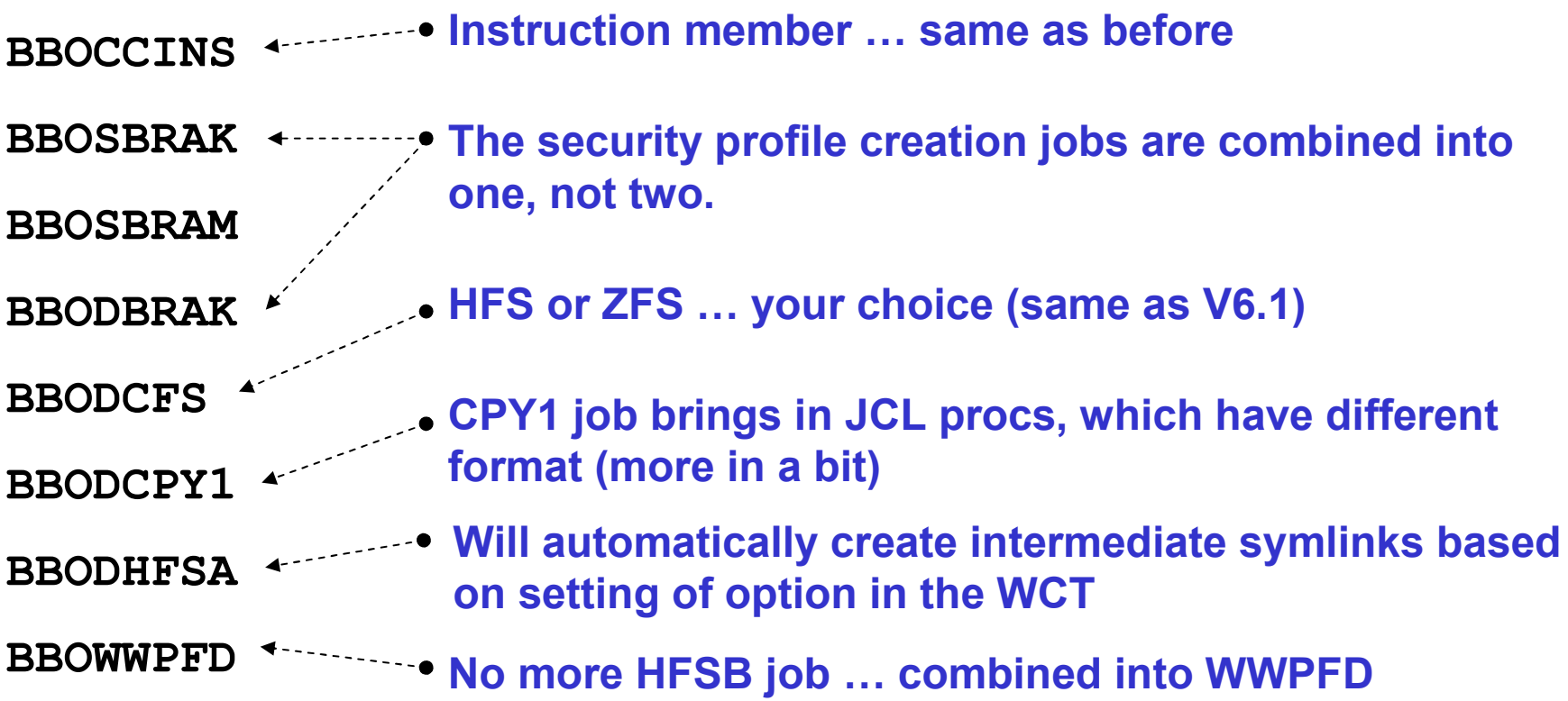
V6.1 Spreadsheet does not help create new server types -- Admin Agent, Job Manager, Secure Proxy, etc.

V7 Edition of the spreadsheet is available – PRS3341

Transition …

Generated Jobs

The jobs are very similar to V6.1, but there are a few notable differences:



Message -- very similar to what we saw in V6.1. Some consolidation of jobs and streamlining. But overall very similar to before.

Transition …

One-Part JCL Start Procedures

Here's an example of the controller JCL start procedure:

```
//B2ACRD  PROC ENV=,PARMS=' ',REC=N,AMODE=00
// SET ROOT='/wasv7config/b2cell/b2noded'
// SET FOUT='properties/service/logs/applyPTF.out'
// SET WSDIR='AppServer'
  :
//*************************************************************
//* Start the Multi-Product PTF Post-Installer                *
//*************************************************************
//APPLY   EXEC PGM=BPXBATCH,REGION=0M,
// PARM='SH &ROOT./&ENV..HOME/bin/applyPTF.sh inline'
  :
//*************************************************************
//* If the RC from the Post-Installer is LE 4 then start      *
//* the WebSphere Application Server                          *
//*************************************************************
//STEP1    EXEC PGM=BPXBATA2,REGION=0M,TIME=MAXIMUM,MEMLIMIT=NOLIMIT,
// PARM='PGM &ROOT./&WSDIR./lib/bbooctlm &AMODE. &PARMS. REC=&REC'
//STDENV DD PATH='&ROOT/&ENV/was.env'
//*
//* Output DDs
//*
//CEEDUMP   DD SYSOUT=*,SPIN=UNALLOC,
//STDOUT    DD SYSOUT=*,SPIN=UNALLOC,
//STDERR    DD SYSOUT=*,SPIN=UNALLOC,
//DEFALTDD  DD SYSOUT=*,S
//HRDCPYDD  DD SYSOUT=*,S
//          ENDIF
//IFTSTEND ENDIF
//*
```

START command still implies passing in ENV= string to uniquely identify server to be started

AMODE=00 a placeholder … 31-bit or 64-bit controlled by contents of configuration XML

SET ROOT= still identifies the configuration HFS mount point

applyPTF.sh still used to bring in maintenance

No longer have IF-THEN to determine if 31-bit or 64-bit module used … one program invoked in either case with contents of XML determining which mode is used

Output DDs now inline with main procedure. No more &INCLUDE or "Z member"

No STEPLIB ... you may add for things like DB2 or MQ … don't need for WebSphere anymore

Transition …

“zDiff Line Items”

Overview of zDiff Items

Background: through feedback at forums such as zBLC, customers have indicated they wish WebSphere on z/OS to exploit the platform strengths.

New SMF 120 Subtype 9 Record

Issue: Previous SMF 120 records were very costly and didn't provide sufficient data, thus rarely used
Solution: New subtype reduces overhead

Thread Hang Recovery

Issue: In the past a hung thread resulted in a servant abend, disrupting operations
Solution: New mechanism that attempts to "shake loose" hung threads

FRCA Caching in Controller

Issue: Caching of objects in WAS cache not as efficient as FRCA, which is handled at lower TCP level
Solution: Exploit FRCA API out of controller. FRCA now used as an external cache of DynaCache

DCS signalling over XCF

Issue: High Availability Manager (HAM) signalling over TCP incurs sizeable overhead. For large server topologies there's an "N-squared" problem -- signalling overwhelms system.
Solution: Exploit XCF with new "plugin" that allows DCS (HAM) signalling over Sysplex facility

All of these are down at the "plumbing" layer and are intended to exploit the platform capabilities more directly. Higher level functionality common across platforms as it should be.

Transition …

SMF 120 Subtype 9, Part 1

A review of the new SMF record …

- **Provides more information about WebSphere transactions to help customers with chargeback information, such as:**
  - **What ran (Application, Servlet, EJB method, MDBean)**
  - **When it ran, how long it took**
  - **Who ran it (Calling host:Port, Security ID: Origin, Received, Invocation)**
  - **Resources used (CPU – CPs, zAAPs, zIIPs, Bytes transferred)**
  - **Other (Classification Names)**
- **Overhead significantly less compared with current SMF Type 120 (subtypes 1-8) records.**
- **A browser to display the contents of the new SMF records is provided.**

Transition …

SMF 120 Subtype 9, Part 2

Feature can be turned on with new variables, or dynamically:

- **Static definition using WebSphere variables:**

```
server_SMF_request_activity_enabled          0 | 1
server_SMF_request_activity_CPU_detail       0 | 1
server_SMF_request_activity_timestamps       0 | 1
server_SMF_request_activity_security         0 | 1
```

- **Dynamically turn 120.9 records on and off, and set the level of details collected, through the MVS Modify (F) command:**

```
F <server>,SMF,REQUEST,[ON | OFF]
F <server>,SMF,REQUEST,CPU,[ON | OFF]
F <server>,SMF,REQUEST,TIMESTAMPS,[ON | OFF]
F <server>,SMF,REQUEST,SECURITY,[ON | OFF]
```

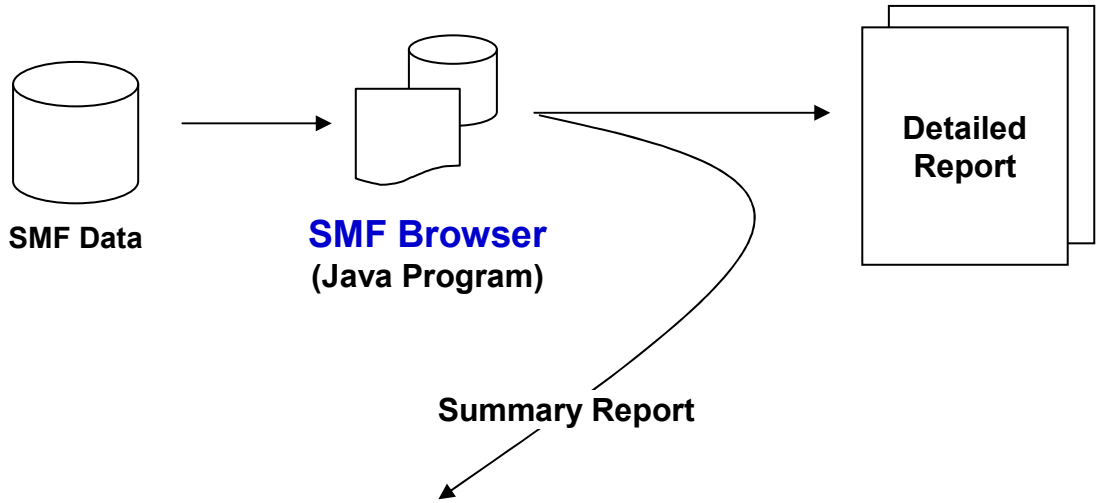
- **DISPLAY command tells you the status of SMF recording within a server**

```
F <server>,DISPLAY,SMF
```

Transition …

SMF 120 Subtype 9, Part 2

The SMF browser will be updated to read and format the SMF data, including the new Subtype 9 record:

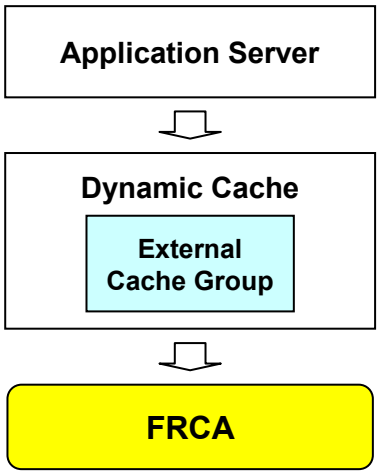



```
SMF 120 Performance Summary V700              Date:  Wed Apr 16 09:10:41 EDT 2008   SysID: SYSB, Page 1

- Record subtypes: 1:Svr_Act. 3:Svr_Int. 5:EJB_Act. 6:EJB_Int. 7:Web_Act. 8:Web_Int. 9:Request
 - subtype 9 Sections: C:CPU, N:Network, Cl:Classification, S:Security, T:Timestamps, U:UserData

SMF -Record Time     Server   Bean/WebAppName      Bytes  Bytes # of El.Time   Enclave_CPU_Time(uSec)
Numbr -Type hh:mm:ss Instance  Method/Servlet      toSvr  fmSvr Call  (msec)     GCP     zAAP     zIIP
1---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0--
    2 120.9  9:10:41 H1SR01B  MyWAR.name/MyServlet  1234   3456   12    7232     490       96       12
    3 120.9  9:10:47 H1SR01B  MyEJB.name/doMethod   1234   3456   12    7232     490       96        7
   23 120.91ClsIIOP-Appl'n EAR-AppName, Jar-ModuleName, EJB-Component Name
```

Transition …

FRCA Caching, Part 1

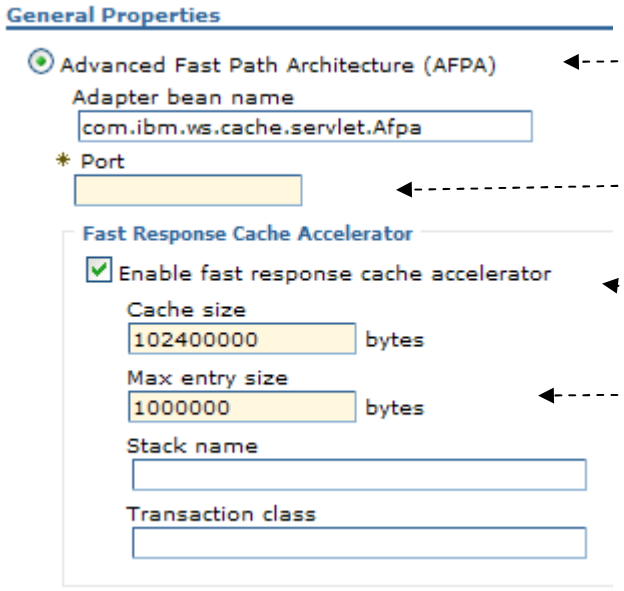
FRCA is a function of TCP, and for years the HTTP Server has exploited it. It's a very good caching mechanism. WAS V7 servers may now exploit it as well.



Exploitation of FRCA is really as an extension to the existing Dynamic Cache ("DynaCache") capability of each Application Server.

FRCA is defined as an "External Cache Group," and the "Adapter Bean" is what provides the function to access FRCA.

Application servers > [server] > Dynamic cache service > External cache group



Enable AFPA (one of four options for external cache member)

Provide a port on which AFPA will listen (this is a net-new port usage)

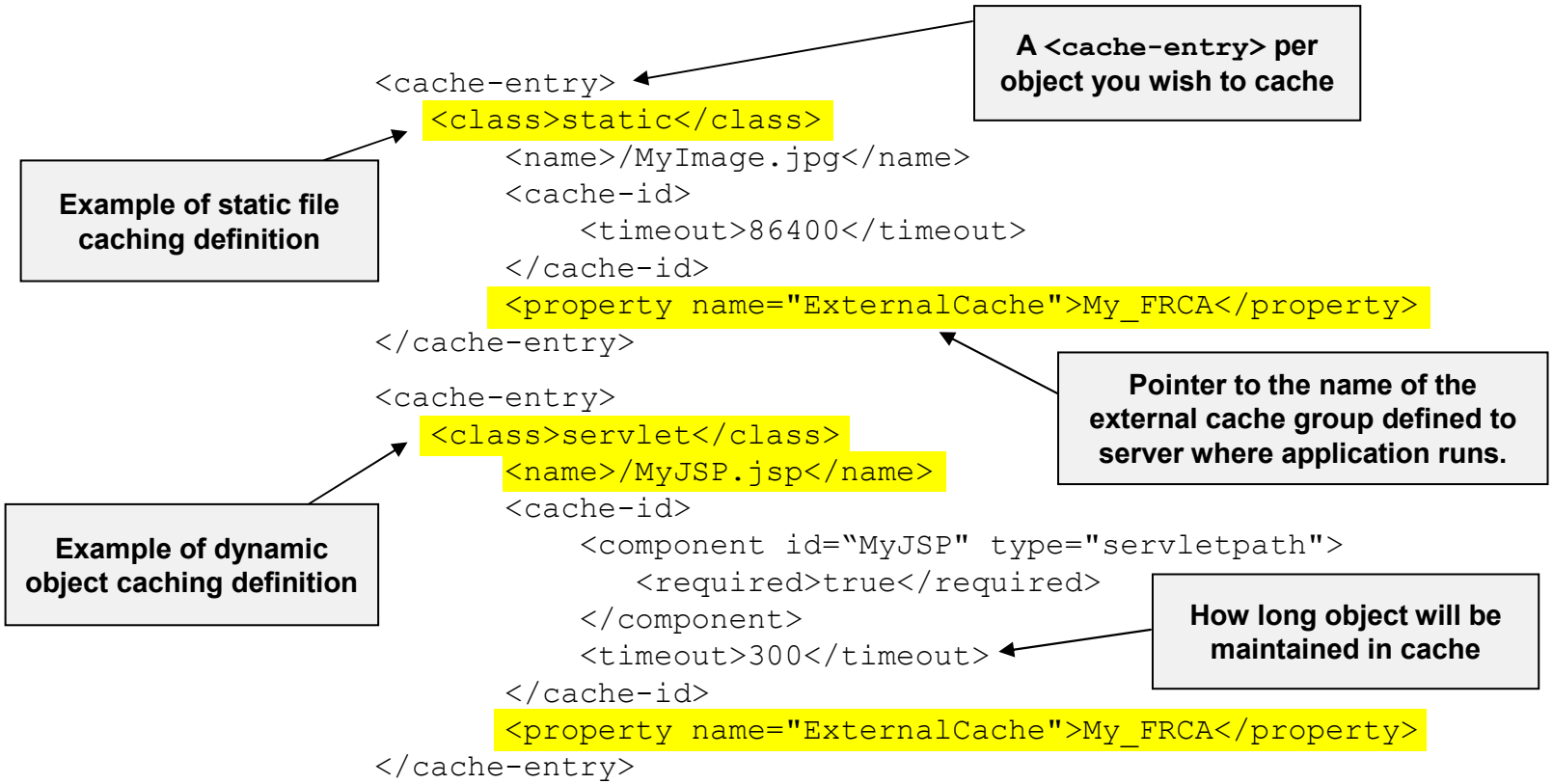
Check the "Enable FRCA" box

Provide FRCA settings

Transition …

FRCA Caching, Part 2

The use of DynaCache (and FRCA below it) is determined by application via the `cachespec.xml` file, which typically resides in the WEB-INF of the WAR:




```
<cache-entry>
    <class>static</class>
        <name>/MyImage.jpg</name>
        <cache-id>
            <timeout>86400</timeout>
        </cache-id>
        <property name="ExternalCache">My_FRCA</property>
</cache-entry>

<cache-entry>
    <class>servlet</class>
        <name>/MyJSP.jsp</name>
        <cache-id>
            <component id="MyJSP" type="servletpath">
               <required>true</required>
            </component>
            <timeout>300</timeout>
        </cache-id>
        <property name="ExternalCache">My_FRCA</property>
</cache-entry>
```

More on `cachespec.xml` file syntax and formatting:

`http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.zseries.doc/info/zseries/ae/`**`rdyn_cachespec.html`**

Transition …

FRCA Caching, Part 3

Few final points:

This is a server-by-server configuration thing
Enabling whole cell implies WSADMIN script that extracts servers and loops through them setting properties.

z/OS 1.9 and above … there is *no* commitment to retro this back to z/OS 1.8

This requires a fix to Communication Server (TCP) in z/OS 1.9. No APAR as yet, but ++USERMOD exists and is under test. APAR to support WAS FRCA on z/OS 1.9 planned.

Remember -- an extension of DynaCache, which means all the features of that (shared cache; pushing to other cluster members) is applicable.
Can get complex quickly, but does not need to be complex for initial validation and PoC testing.

With USERMOD don't need SERVAUTH profile for `EZB.FRCAACCESS.*.*` … can code if you want, but not required.
Don't know if APAR will call for same, but will likely be the same

Transition …

DCS/XCF, Part 1

Before we explain this function, a review of some essential concepts may prove helpful.

High Availability Manager

Function of WebSphere Application Server (all platforms) that provides:

- **A way for singleton services to be made HA. Examples: transaction managers for cluster members and the default messaging provider**
- **Data exchange mechanism (the "bulletin board")**
- **Messaging infrastructure for things like Domain Replication Service**

A HAM "instance" runs on every server in a "core group," with one designated as "core group coordinator"

DCS

Stands for "Distribution and Consistency Services" … essentially a set of messaging and signaling protocols within a WebSphere Application Server cell.

This new function is a mapping of DCS over a different transport -- XCF rather than TCP

Core Group

A collection of nodes (and servers in those nodes) to form the boundary of HAM coordination and DCS signaling. By default one core group -- `DefaultCoreGroup`.

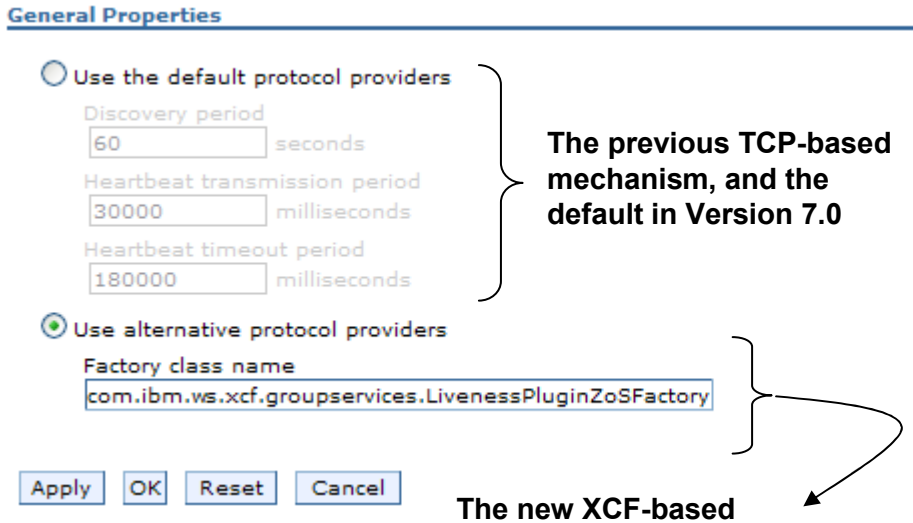
One strategy for containing excessive DCS overhead is segmenting cell into multiple core groups.

Transition …

DCS/XCF, Part 2

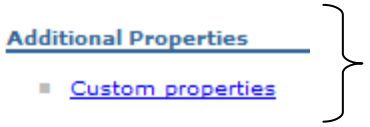
The DCS signalling over TCP can be resource-intensive, limiting the size of WAS topologies. This feature allows DCS signalling over XCF.

Enabling this is relatively simple … it's a property of the "Core Group" settings:



The previous TCP-based mechanism, and the default in Version 7.0

The new XCF-based protocol provider, with the XCF factory class specified



Custom settings

`IBM_CS_STACK_CHECK_INTERVAL_SECS`

Determines how frequently the alternate protocol provider checks the liveness of a core group member. Default = 30 seconds.

`IBM_CS_STACK_CHECK_FAILS`

Determines how many times the alternate protocol provider attempts to contact a core group member can fail before the alternate protocol provider notifies the high availability manager that a member is not active

Transition …

DCS/XCF, Part 3

Some restrictions to this new functionality:

Only works with WAS V7.0 or higher

- **WAS V7.0 supports a "mixed-level" cell, but a core group using DCS/XCF must be all V7.0 or higher.**
- **Keep this in mind if you look to enable during migration … whole core group has to come up to V7.0 before you can enable DCS/XCF.**

DCS/XCF requires that all nodes in a core group operate on z/OS

- **Core groups are ideally "homogenous" but it is possible to create mixed core groups**
- **DCS/XCF function requires that all members of core group participate using same exchange transport. Since XCF is z/OS only, it means core group must be all-z/OS.**
- **If you want to merge in a node using, say, zLinux, then you'd have to revert core group transport back to "default protocol provider"**

Transition …

Thread Hang Recovery, Part 1

In the past if a timeout value expired because of a "hung" thread, the servant region was abended. New function attempts to "shake loose" thread.

Request

1. Request received
2. Timer value set
3. Thread dispatched
4. Time pops
   a. Pre-V7 -- Abend Servant with EC3
   b. V7.0 -- Attempt "Thread Hang Recovery"

If the thread *can* be freed, then it is and the user request is terminated
(with a choice of what "dump action" is taken -- none, svcdump, javacore, heapdump, and traceback)

If the thread *can not* be freed, then other processing occurs:

- **Servant may still abend**
  (if settings give WebSphere no other options)
- **Or the thread may be left in hung state if threshold value not exceeded**
  (a new optional variable sets a percentage of hung threads that causes abend of servant)
- **Thread may be left in hung state if last "surviving servant"**
  (a new optional variable prevents last servant from abending when minimum servants more than one)

Transition …

Thread Hang Recovery, Part 2

New variables and DISPLAY commands

server_region_<*type*>_stalled_thread_dump_action

Specifies what WAS will produce if a stalled thread can't be interrupted. Values are none, svcdump, javacore, heapdump, and traceback. <type> is the access method -- http, https, sip, sips, iiop and mdb.

server_region_stalled_thread_threshold_percent

Specifies a percentage of hung threads that must exist before the servant region will be recycled.

servant_region_custom_thread_count

Works in conjunction with a new ORB thread setting of CUSTOM, this allows you to set the number of threads available to each servant.

control_region_timeout_save_last_servant

Works in conjunction with a new ORB thread setting of CUSTOM, this allows you to set the number of threads available to each servant. (Previous settings of ISOLATE, IOBOUND, LONGWAIT and CPUBOUND still exist)

server_region_request_cputimeused_limit

Specifies a maximum CPU time for any given thread in the server. This prevents runaway programs. WAS will interrupt the thread that consumes more time than is permitted by this setting.

server_region_cputimeused_dump_action

Specifies the dump action WAS will take if the CPU time value is exceeded. Values are the same as the stalled thread dump action at the top of this page.

```
F <controller>,DISPLAY,THREADS,ALL
F <controller>,DISPLAY,THREADS,TIMEDOUT
F <controller>,DISPLAY,THREADS,REQUEST=value
F <controller>,DISPLAY,THREADS,ASID=value
F <controller>,DISPLAY,THREADS,AGE=value
```

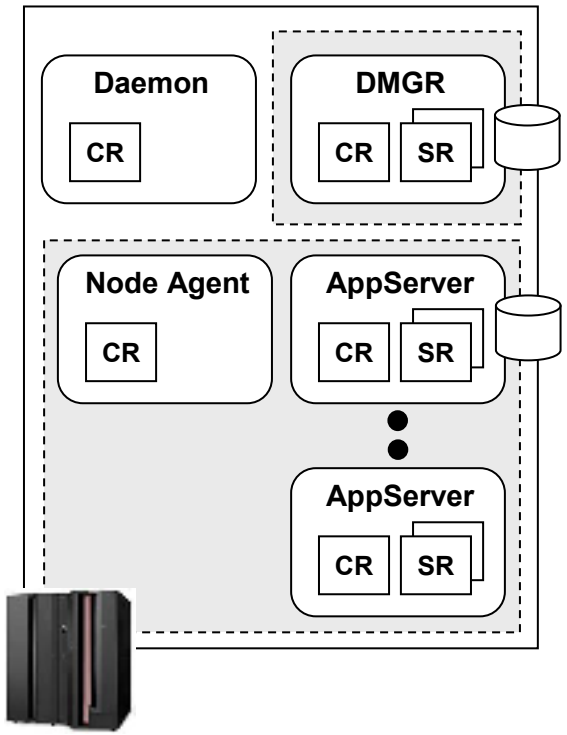
Displays threads for a server (controller and all servants) and provides information about the state of those threads - in Java Wait, Hung, etc.

Transition …

Miscellaneous

Resource Utilization and Performance

Three key pieces to this:



Memory footprint

CPU utilization

Server startup time

Transition …

Handy New Things

An unsorted listing of other things … no particular order

Exploitation of WLM Parallel Servant startup - you can tell us to tell WLM to start the 2->N min-SRs concurrently. Comes up faster if you have the system resources to support it.

Modify command to dynamically alter min/max SRs without restarting the server.

z/OS provided support to re-use ASIDs in z/OS 1.9. If you always start your controllers manually you can exploit this now. If you let WAS start them (via admin stuff) then you have to wait for us to update our generated start commands. Should be coming in V6.1 via a PTF and in V7 (maybe in a PTF after GA).

Transition …

Migrating to V7 from Prior Releases

We've not *yet* done an actual migration … but here's our outlook:

Process is similar as before:

- Use WCT (zPMT) to configure migrations jobs (ISPF panels no longer an option)
- Migrate node-by-node, with DMGR the first node migrated
- Servers in node must be stopped at the time of migration
- Federated application server nodes require DMGR to be up at time of migration

Supported Migration Map

- V5.0 ⇨ V7.0 **Unsupported**
- V5.1 ⇨ V7.0 **Supported**
- V6.0 ⇨ V7.0 **Supported**
- V6.1 ⇨ V7.0 **Supported**

Note: there may be minimum maintenance level requirements a node must be at before migration will work. Specifics not yet available, but look for them at the time of migration.

Mixed Nodes Allowed

- V7.0 DMGR is capable of supporting downlevel (V6.1, V6.0 and V5.1) nodes
- Some restrictions may apply in terms of what a V7.0 DMGR can create or do in a downlevel node. Watch for those restrictions spelled out in later migration documents

Construction of new V7.0 cell in parallel to V5.x or V6.x cell always an option

- We've seen some who opt to start fresh and rebuild a new-new cell, which avoids migration
- This implies performing all the post-creation customization work, of course

Transition …

Questions

That's the end of the presentation format

Now Questions/Answers and Discussion