Cheat Sheet #77 zTidBits z/OS BCPii

•IBM provides support within z/OS that allows authorized applications to query, change, and perform operational procedures against the installed System z hardware base through a set of application program interfaces.
- These applications can access the System z hardware that the application is running on and extend their reach to other System z processors within the attached process control (Hardware Management Console) network.

•Using the **Base Control Program internal interface (BCPii),** an authorized z/OS application can perform the following actions:
- Obtain the System z topology of the current interconnected Central Processor Complexes (CPCs) as well as the images, capacity records and activation profiles on a particular CPC.
- Query various CPC, image (LPAR), capacity record, and activation profile information.
- Set various configuration values related to CPC, image and activation profiles.
- Issue commands against both the CPC and image (LPAR) to perform minor or even significant hardware- and software-related functions.
- Listen for various hardware and software events that might take place on various CPCs and images throughout the HMC-connected network.

•Communication to the Support Element (SE) / Hardware Management Console (HMC) using BCPii is done completely within the base operating system and therefore *does not require* communication on an IP network (intranet) for connectivity, providing complete isolation of your System z hardware communication from any other network traffic within the intranet/internet.

•Calls using the BCPii Application Programming Interfaces (APIs) can be made from either C or assembler programming languages.

BCPii setup and installation

•Before an installation begins to issue BCPii APIs, a series of setup and installation steps must be performed. A summary of these steps is listed below. For additional details on each of these steps, see the supporting documentation that explains how each of these steps is accomplished:

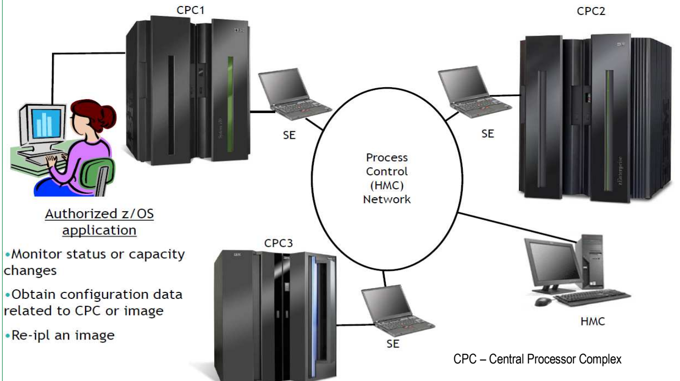
1. Configure the local Support Element (SE) to support BCPii:
   a. Check the levels of hardware that BCPii supports.
   b. Enable cross-partition authority for each image (LPAR) that you want to grant BCPii access.
   c. Define an uppercase BCPii SNMP community name on the SE.
2. Authorize an application to use BCPii, including authority to specific resources(such as CPCs, images and capacity records):
   a. Check that the BCPii application is program-authorized.
   b. Check that the BCPii application has general authority to use BCPii.
   c. Authorize the BCPii application to access the particular resource that requires BCPii service.
   d. Define an uppercase BCPii SNMP community name in the security product for each CPC as it was defined on the SE.
      - Use the APPLDATA field with the CPC profile definition to associate a BCPii SNMP community name with a particular CPC.

   These steps enable communication to the local CPC and allows the BCPii address space to initialize.
3. Configure the BCPii address space.
4. If the caller is running in a z/OS UNIX System Services environment, set up the notification mechanism to allow hardware and software events to be propagated to the z/OS UNIX application.

•After you have activated the BCPii address space, you need to know how to control the address space.

Using BCPii ; a Simple Example:

–My company monitor application wants to keep track of the list of CPCs and how many CBU test activations are remaining on each CPC, the list of images (LPARs) on those CPCs, what operating systems are running on all those images, as well as the OS levels and if z/OS, what sysplex are they a part of.



•Monitor status or capacity changes

•Obtain configuration data related to CPC or image

•Re-ipl an image

CPC – Central Processor Complex

Examples of information you can query:

–CPC information

•General information

–Name, serial, machine type, id, networking information

•Status information

–Operating status and other status values

•Capacity information

–Various CBU info, Capacity on Demand info, Processor configuration, including IFA, IFL, ICF, IIP

•Power savings information *(available on zEnterprise hardware only with APAR OA34001 on V1R10 - V1R12)*

–Is power savings available?, current power save mode, supported power saving modes available

–Image information

•General information

–Name, OS info

•Capacity information

–Defined capacity, Processor weights

Setting up connectivity to the support element

•BCPii uses a low-level operating system connection to establish communication between an authorized application running on a z/OS image (LPAR) and the Support Element (SE) associated with the Central Processor Complex (CPC) that contains this z/OS image. You must configure the support element to permit these BCPii communications if BCPii services are required to be available by your installation.

Note: In order to customize the API settings controls on the SE, your system administrator must have administrator rights to access these panels.

Levels of hardware BCPii supports

•The HWIBCPii address space, which supports the issuing of BCPii APIs from a z/OS image, will run on any level of hardware that supports the level of the z/OS operating system in which BCPii is included. However, there will be reduced BCPii functionality when targeting any system with a BCPii request which is not running on a z196 or z10 with all the recommended microcode levels installed.
- The further the hardware level is from a z10, the greater the restrictions.
- BCPii applications might need to perform hardware or software functions on CPCs other than the CPC on which the application is running.
- Such requests can be targeted to other System z hardware at a lower or higher hardware level than the local CPC, provided that these hardware levels are supported to coexist with the local CPC level.

  Note: Each version of hardware has subtle or sometimes significant changes in the way information is displayed and saved in the support element.

The BCPii address space functions:

•The bridge between a z/OS application and the support element

-Manages all application connections

-Builds and receives all internal communication requests to the SE

-Provides an infrastructure for storage required by callers and for the transport communicating with the SE

-Provides diagnostic capabilities to help with BCPii problem determination

-Provides security authentication of requests

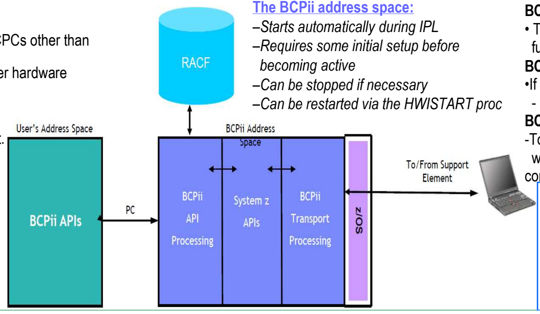
The BCPii address space:

–Starts automatically during IPL

–Requires some initial setup before becoming active

–Can be stopped if necessary

–Can be restarted via the HWISTART proc



Enable BCPii communications on the support element

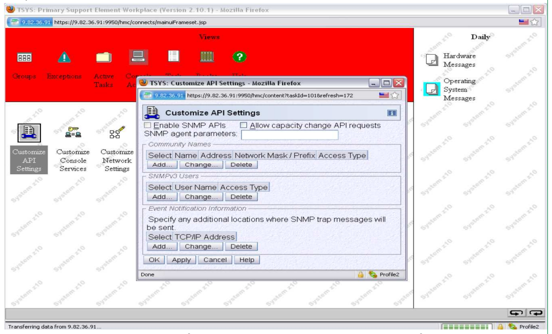
•You need to enable cross-partition authority on the support element to allow the support element to accept the BCPii APIs flowing from the user application through the HWIBCPii address space.
- This setting controls whether a logical partition can issue a subset of control program instructions to other logical partitions activated on the same CPC.

Note: This setting must be selected on the local SE associated with the CPC of the image that the z/OS BCPii application is running on. It must also be selected for any other system for which BCPii communication is required

Define the BCPii community name on the support element

• BCPii uses an SNMP community name to provide a level of security between the z/OS image that is executing the BCPii service and the support element itself.
- An SNMP community is a logical relationship between an SNMP agent and an SNMP manager.
- The community has a name, and all members of a community have the same access privileges: they are either read-only (members can view configuration and performance information) or read-write (members can view configuration and performance information, and also change the configuration).

  NOTE: Failure to set this properly on the local SE associated with the image of z/OS that is running BCPii results in a severe BCPii failure and you cannot start the address space.


TSYS: Primary Support Element Workplace (Version 2.10.1) - Mozilla Firefox
9.82.36.91 https://9.82.36.91:9950/hmc/connects/mainuiFrameset.jsp
Views
Groups
Exceptions
Active Tasks
Daily
Hardware Messages
Operating System Messages
Customize API Settings
Customize Console Services
Customize Network Settings
TSYS: Customize API Settings - Mozilla Firefox
9.82.36.91 https://9.82.36.91:9950/hmc/content?taskId=101&refresh=172
Add...
Change...
Delete
OK
Apply
Cancel
Help
Done
Profile2
Transferring data from 9.82.36.91...

Customize API Settings

□ Enable SNMP APIs □ Allow capacity change API requests

SNMP agent parameters:

Community Names

Select	Name	Address	Network Mask / Prefix	Access Type

SNMPv3 Users

Select	User Name	Access Type

Event Notification Information

Specify any additional locations where SNMP trap messages will be sent.

Select	TCP/IP Address

Setting up authority to use BCPii

•Given the nature of the BCPii APIs and the capabilities of a BCPii application to potentially modify vital hardware resources, a number of authority validations are performed for each BCPii requestor.
- A BCPii application needs to have program authority, general security product authority to be able to issue BCPii commands, authority to the particular resource that the application is trying to access, and a community name defined in the security product for each CPC to which communication is required.

Authority to the particular resource

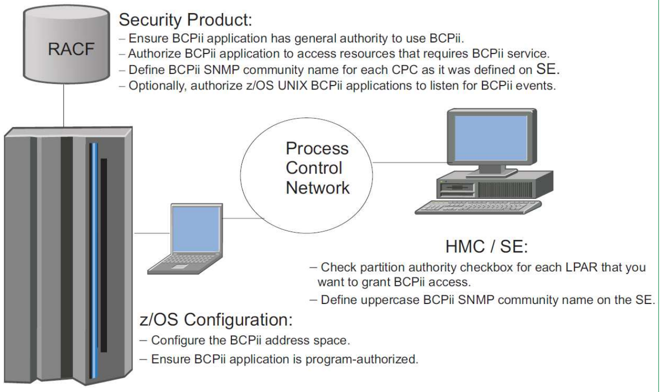
•A BCPii application needs to have authority to the particular resource that it is trying to access. That particular resource can be the CPC itself, an image (LPAR) on a particular CPC, or a particular capacity record on a particular CPC. BCPii needs a profile defined in the FACILITY resource class that represents the target of the particular BCPii request. The profile name required to be defined depends on the type of the particular resource required.

Setting up event notification for BCPii z/OS UNIX applications

•Applications running in a started procedure, batch, TSO or other non z/OS UNIX environment can use the HWIEVENT service and provide their own ENF exit that receives control when the application-requested events occur on the target CPC or image.
- Applications running in a z/OS UNIX environment do not have normal ENF exit processing capabilities available and cannot readily listen for ENF signals.
- The Common Event Adapter (CEA) address space allows z/OS UNIX applications to be able to receive such event notifications. BCPii provides services that use the CEA functionality to deliver these same events to z/OS UNIX callers.

CEA address space setup

•The Common Event Adapter (CEA) address space must be active to allow the z/OS UNIX-only services of BCPii to operate.
- CEA has two modes of operation: minimum or full-function mode.
- If the z/OS UNIX-only services of BCPii are required to be available, CEA must be running in full-function mode.



Security Product:
– Ensure BCPii application has general authority to use BCPii.
– Authorize BCPii application to access resources that requires BCPii service.
– Define BCPii SNMP community name for each CPC as it was defined on SE.
– Optionally, authorize z/OS UNIX BCPii applications to listen for BCPii events.

HMC / SE:
– Check partition authority checkbox for each LPAR that you want to grant BCPii access.
– Define uppercase BCPii SNMP community name on the SE.

z/OS Configuration:
– Configure the BCPii address space.
– Ensure BCPii application is program-authorized.

NOTE: Communication to the Support Element (SE) / Hardware Management Console (HMC) using BCPii is done completely within the base operating system and therefore does not require communication on an IP network (intranet) for connectivity, providing complete isolation of your System z hardware communication from any other network traffic within the intranet/internet.

BCPii startup and shutdown

• The BCPii address space normally does not need to be started or shut down. BCPii initialization occurs during system IPL. If the configuration is correct, no further action is required. The address space remains active and ready to handle BCPii requests.

BCPii address space at IPL

•If the HWIBCPii address space is not active after an IPL has been done, look for HWI* messages in the system log.
- Most of the time, these messages pinpoint the reason for the failure of BCPii to become active.

BCPii commands, because of the very nature of what they are attempting to do, may take a significant amount of time to complete.

-To prevent applications from being tied up for an excessive amount of time while waiting for the command to complete, HWICMD will return to the caller either when the command has been *accepted* by the target support element (SE) or when the command was found to contain errors. The actual completion of the command can be determined by consulting the final return code returned in the BCPii command response event.

What is z/OS BCPii vs. BCPii mentioned in TSA?

•Tivoli System Automation (ProcOps) allows its automation product to use one of 2 transport protocols:

–SNMP over an IP network

–BCPii protocol (internal transport)

•TSA's BCPii implementation is similar but not z/OS BCPii and requires TSA, Netview and Comm Server.

•BCPii transport in TSA is for TSA usage only

•z/OS BCPii can run in ANY address space and has no other product requirements.