z/OS V1R11 Communications Server
SNA and EE

z/OS Communications Server Development, Raleigh, North Carolina

This presentation describes the enhancements to the Communications Server in z/OS V1R11 for the SNA and enterprise extender (EE) theme which includes enhancements to the SNA component of the z/OS Communications Server.

SNA and EE

- HPR performance enhancements
  - Progressive ARB mode
  - Path switching
  - More enhancements
- APPN topology database update enhancements
- Display potential model application name
- Provide ACF/TAP as part of z/OS Communications Server
- Include data space VIT with INOP dump

The main enhancement in this release for SNA and EE is a new high performance routing (HPR) adaptive rate-based (ARB) protocol. Progressive ARB mode addresses performance issues seen in environments where distributed platforms are implemented in virtualized systems. There are also HPR enhancements with path switching and additional enhancements.

APPN topology database update enhancements help diagnose TDU wars.

z/OS V1R11 Communications Server enhances the DISPLAY MODELS command to identify which application model definition is used to build a dynamic application definition. Customers can use this capability to prevent problems with dynamic application definitions being built incorrectly.

ACF/TAP is now included with z/OS Communications Server.

VTAM® INOP dump processing is enhanced to automatically capture the VIT data space (ISTITDS1) in the dump when the VIT data space is in use.

Progressive Mode ARB implemented

- Progressive Mode ARB is an adjustment to Responsive Mode ARB
  - When ARB Slowdown messages received
    - Amount of data awaiting acknowledgment is checked
  - If amount of data is small, Slowdown messages treated as Restraints
    - Slowdowns normally trigger transmission rate cuts, restraints do not
- This solution is limited to one hop HPR pipes over EE
  - Includes a two hop HPR pipe through a VRN

- PTFS for OA26490 are available for prior releases
  - Ensures that prior releases of VTAM don't regress to using Base Mode ARB when they see the new ARB mode bits set

ARB (Adaptive Rate-Based) is a flow control mechanism used by HPR. The original introduction of HPR had an ARB mode known as Base Mode ARB. When Enterprise Extender (EE) was shipped, a second ARB mode, known as Responsive Mode ARB was developed. Most customers use Responsive Mode ARB. In its effort to proactively prevent congestion, the ARB flow control mechanisms have always been very sensitive to minor variations in round-trip time or unpredictability in response time from the partner. Frequent swings in round-trip times lead to ARB rate cuts that prevent the RTP pipe from ramping up to speeds consistent with what the media and network should allow. This results in degraded throughput and reduced response time.

To address the problems associated with the variation in responsiveness of a constrained environment, the basic approach that is being taken with this solution is to "sanity check" any Slowdown messages received. When VTAM's RTP code receives Slowdown messages in an ARB segment, RTP will check the amount of outbound data that is awaiting acknowledgment. If that number is small, RTP will ignore the SLOWDOWNs and will treat them as RESTRAINTs. So, VTAM will not slow down its transmit rate in this case. This will make VTAM code less reactive to clock issues at the partner RTP.

This solution is for one hop HPR pipes. When the phrase, one hop HPR pipes, is used, that includes two hop pipes that go through a VRN.

There are PTFs available to make sure that prior releases of VTAM (V1R8-V1R10) don't go back to using Base Mode ARB when they see an ARB mode that they don't recognize in the various flows.

Using progressive mode ARB

- HPREEARB=PROGRESS can be specified in three major nodes
    - On the PU in a Switched Major Node
    - On the connection network GROUP in an XCA Major Node
    - On the EE model PU in a Model Major Node

```
>----+------------------+--->
     '-HPREEARB=PROGRESS-'
```

- Issue Display NET,ID=rtp_pu to see the ARB mode being used

```
D NET,ID=CNR00004
IST097I DISPLAY ACCEPTED
IST075I NAME = CNR00004, TYPE = PU_T2.1
. . .
IST2267I RTP PACING ALGORITHM = ARB PROGRESSIVE MODE
. . .
```

VTAM provides a new parameter, HPREEARB, that the system programmer can use to define which EE connections they prefer to use the new ARB mechanism. This HPREEARB parameter can be specified on the switched PU for predefined connections, and on the XCA GROUP for connection network-based PUs. Additionally, the parameter can be specified on a DYNTYPE=EE model PU.

If Progressive Mode ARB is in use by an RTP pipe, a display of the pipe will include IST2267I. IST1697I is displayed if Responsive Mode ARB is in use. The absence of both messages means that Base Mode ARB is being used.

HPR path switch processing can be too sensitive
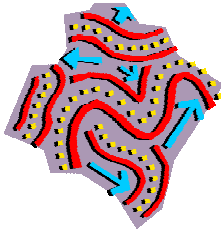
- An HPR path switch is an effort to improve reliability
- However, at times the HPR path switch logic is too sensitive
  - Transient network conditions cause temporary swings in round trip time (RTT) which can cause a path switch
  - HPR path switches back to same route
    - Consumes processor cycles and produces unwanted messages
- Unresponsive partner example:
  - CP-CP RTP pipe enters path switch after three unsuccessful retries
    - Minimum SRQ timer value is 250ms
  - CP-CP pipe can enter path switch in roughly one second
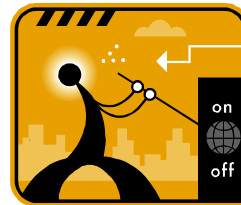
An HPR path switch occurs automatically to bypass link and node failures if an acceptable alternate path is available. During a path switch, a new route is calculated in an effort to bypass the failure.

In its effort to improve reliability, the HPR path switch logic at times is too reactive to minor variations in round-trip time. Transient network conditions can cause a path switch. In some cases, the newly calculated route is the same as the current route the pipe is using. In these cases, the path switch is unnecessary and unwanted. This consumes processor cycles and can produce a very significant amount of message output that is not particularly useful.

You will see an example of a path switch for a CP-CP RTP pipe. In this case, the partner CP-CP RTP endpoint is unresponsive. The minimum SRQ timer value supported by z/OS Communications Server is 250ms. So, for a CP-CP RTP pipe, which has an unresponsive partner, path switch is entered after three unsuccessful retry attempts. This means that a CP-CP pipe can enter path switch in a period of time equal roughly to one second.

New options for delaying start of HPR path switches

- New controls will allow you to specify an HPR path switch delay value
  - Specifies the minimum amount of time a z/OS CS RTP endpoint must wait before initiating a path switch due to an unresponsive partner
  - Does not control path switches initiated due to a TG INOP, MODIFY RTP command, or the PSRETRY function

- Solution is available for all RTP pipes with endpoints in this z/OS CS node
  - Only controls the path switch logic on this side of the pipe
  - Path switch delay value is not negotiated with the RTP partner

To address these problems, new controls are available to allow you to specify an HPR path switch delay value. This delay will allow for a minimum amount of time that a z/OS CS RTP endpoint must wait before initiating path switch due to an unresponsive partner. This new keyword does not have any control over path switches initiated due to TG INOP, F RTP command, or the PSRETRY function.

This solution is available for all RTP pipes originating in this z/OS CS node. This solution only controls the path switch logic on this side of the RTP pipe. The path switch delay value is not negotiated with the RTP partner. It is not required to have both RTP endpoints running with the HPR path switch delay function. However, if you want the RTP partner node to support the path switch delay function, you can investigate if this function is supported on the partner platform and implement the necessary changes if applicable.

snaee.ppt

How to specify the delay time for HPR path switches

- New start option (HPRPSDLY) specifies time to wait

```
            .-HPRPSDLY=0------------.
  >>----+----------------------+---><
            '-HPRPSDLY=ps_delay------'
```

- HPRPSDLY can also be specified in three major nodes
  - On the PU in a Switched Major Node
  - On the connection network GROUP in an XCA Major Node
  - On the EE model PU in a Model Major Node

- EEDELAY represents a delay value sufficient to allow the EE Keep-Alive mechanism to INOP the connection should connectivity be lost

```
        .-HPRPSDLY=value_of_HPRPSDLY_start_option--.
  >>----+------------------------------------------+---><
        '-HPRPSDLY=+-ps_delay-+------------------'
                   '-EEDELAY--'
```

A new start option allows you to control the minimum amount of time a z/OS RTP endpoint waits before initiating a path switch due to an unresponsive partner. The range for this new start option is from 0 to 240 seconds. The default is zero which means an RTP pipe will enter path switch when a predetermined number of retry attempts have been unsuccessful without regard to how long the retry attempts took. This start option can be modified with the F VTAMOPTS command.

VTAM also provides the HPRPSDLY parameter on three major nodes. This HPRPSDLY parameter can be specified on the switched PU for predefined connections, and on the XCA GROUP for connection network-based PUs. HPRPSDLY can also be specified on a DYNTYPE=EE model PU.

HPRPSDLY allows the system programmer the granularity to customize path switch delay values for specific EE connections. The default for this parameter is the value of the HPRPSDLY start option. This parameter accepts the same values as the HPRPSDLY start option, plus one extra value, EEDELAY. When EEDELAY is specified, VTAM calculates the delay value to allow enough time for the EE Keep-Alive mechanism to INOP the connection should connectivity be lost to the partner. In the event an EE connection suffers an INOP, all local RTP pipes utilizing this link automatically path switch due to the link INOP condition. The EEDELAY option can be beneficial in configurations where EE is the only path to the partner. In these environments, there are no alternate routes available so path switching is unproductive. This can avoid unnecessary route calculations in addition to reducing console output for path switch processing.

Example of HPRPSDLY=EEDELAY

- Display NET,EE see the EE Keep-Alive timer values:

```
D NET,EE
IST2000I ENTERPRISE EXTENDER GENERAL INFORMATION
IST1685I TCP/IP JOB NAME = TCPCS
IST2003I ENTERPRISE EXTENDER XCA MAJOR NODE NAME = XCAIP
IST2004I LIVTIME = (10,0)     SRQTIME =     15  SRQRETRY =      3
```

- Display NET,ID=rtp_pu,HPRDIAG=YES to see the calculated path switch delay value:

```
D NET,ID=CNR00004,HPRDIAG=YES
IST097I DISPLAY ACCEPTED
IST075I NAME = CNR00004, TYPE = PU_T2.1
. . .
IST924I ------------------------
IST1984I PATH SWITCH INFORMATION:
IST2271I PATH SWITCH DELAY = 90
```

This slide has an example where HPRPSDLY=EEDELAY is specified for the EE connection. In this example, CNR00004 is an RTP pipe which is utilizing this EE connection.

If you issue the Display EE command, you can see the EE Keep-Alive timer values in message IST2004I. If you issue the Display ID command, specifying the HPRDIAG=YES option, you can see the path switch delay value that VTAM calculated. In this example, message IST2271I shows the RTP pipe has a calculated value of 90 seconds. This means this end of the RTP pipe will not path switch due to an unresponsive partner until 90 seconds has passed. During this time, VTAM will continue to periodically try to contact the partner.

snaee.ppt

Additional EE/HPR improvements

- VTAM will free packets for fully acknowledged PIUs, regardless of whether gaps exist in the acknowledgments
    - Previously, VTAM only freed PIUs acknowledged in order
    - This will reduce any storage spikes associated with dropped packets
- The amount of ECSA storage required for each HPR pipe reduced

- EE IPSEC performance improvements
    - Use zIIP processors more effectively for inbound and outbound traffic
    - Send smaller traffic bursts to minimize retransmission likelihood
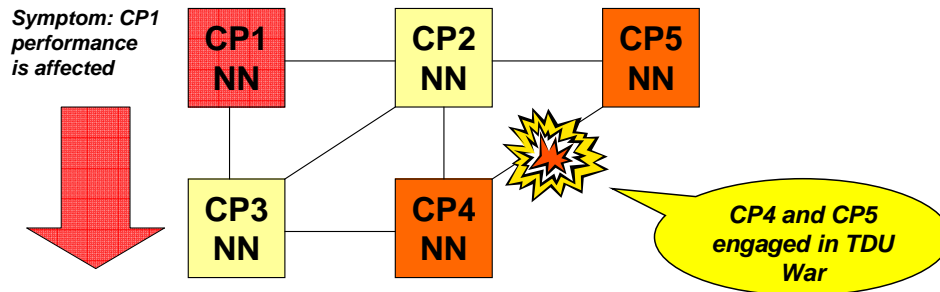
The HPR architecture allows all acknowledged packets to be freed, regardless of the order in which they arrived at the partner. Previously, VTAM did not make use of this feature, and only freed packets when they were acknowledged in order. VTAM will now acknowledge packets, even those after a gap in the acknowledgements, which allows that storage to be freed. This means that if there is a dropped packet that is followed by several megabytes of data, when the Status Segment is received by VTAM reporting the gap, VTAM will free the packets acknowledged after the gap. This will reduce the storage held by VTAM while the dropped packet is retransmitted. Note that VTAM will still only free packets once all of the packets for a PIU have been acknowledged.

In addition to reducing the storage requirements associated with gap processing, VTAM has also reduced the ECSA storage required for each RTP pipe by relocating nonessential ECSA control blocks to VTAM's private region.

Changes have also been made to improve the performance of EE traffic when IPSEC is enabled. One change is that z/OS Communications Server exploits zIIPs for inbound and outbound traffic patterns as efficiently as possible while minimizing latency overhead. A second change breaks large bursts of IPSec traffic being sent across EE into smaller four-segment batches. This allows the smaller batches to be filtered/encrypted and sent to the remote node before subsequent smaller bursts, avoiding out-of-order retransmission scenarios on the remote node.

snaee.ppt

TDU wars can be challenging to diagnose

- Continuing improvements to APPN topology component processing have slightly increased the possibility of TDU wars

- Determining which APPN nodes are at fault not always easy

  – Node undergoing performance degradation might be innocent victim, not perpetrator

Symptom: CP1 performance is affected

CP1 NN   CP2 NN   CP5 NN

CP3 NN   CP4 NN

CP4 and CP5 engaged in TDU War

Recent z/OS Communications Server releases have increased the sophistication of the APPN topology component, and have improved topology awareness across the APPN network. With the increased complexity in APPN topology it is possible that situations have been introduced where, under certain unusual error conditions, TDU wars might occur.

A major concern with TDU Wars, even beyond the performance impact, is that it is very challenging to determine the root cause of the TDU War. The biggest challenge is that the node experiencing the performance issues is often not one of the nodes that is generating the topology updates flooding the network. In this example, the resource in question is the TG between CP4 and CP5. CP4, which in this example is the true owner of the TG, initially sends out a TDU that reports the TG as operative. What if CP5 erroneously believes it has more current information about this TG, for instance that the TG is inoperative? When CP5 gets the update from CP4, CP5 will generate a new TDU saying the resource is inoperative. CP4 responds with a new TDU, correcting CP5's information, the TDU War is on, and there is no end in sight. Meanwhile, CP1 is being flooded with TDU flows providing "updates" for this TG. Eventually CP1 starts to suffer performance degradation. The normal response is to get traces and dumps at CP1, but the information obtained sheds no light on the fact that CP5 is the node in error.

Enhance topology updates to identify node updating resource

- Define Topology Resource Sequence Number Update (x'4E') control vector on TDU flows
  - Can be appended to any TDU that contains a node update or TG update
  - Can identify last two nodes that generated updates about this resource
- New start option available to control when control vector is included
  - Specify ALWAYS to include it on all TDU flows
  - Specify NEVER to exclude it from all TDU flows
  - Specify resource_threshold to include x'4E' only if RSN is updated that often within a 24-hour interval

```
      .-TDUDIAG=1000-----------------.
 >>--|-------------------------------|--><
      '-TDUDIAG=_ _resource_threshold-|
                     |_ALWAYS------------|
                     |_NEVER-------------'
```
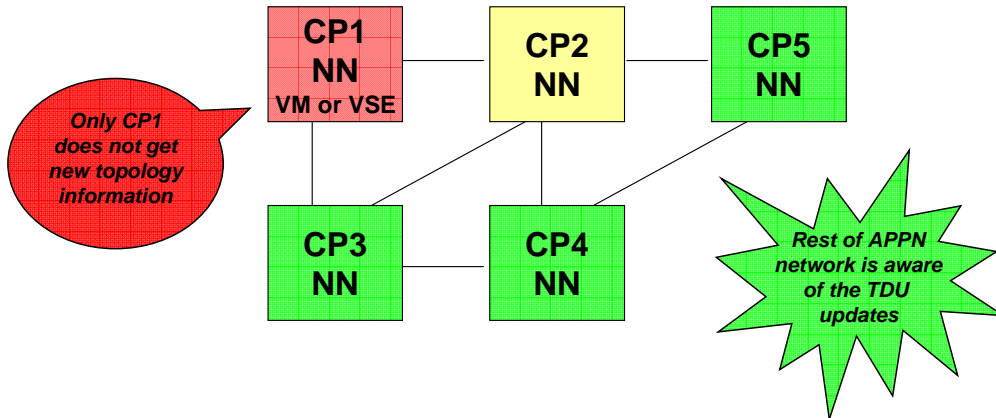
To get better diagnostic information at the impacted node (CP1 in the previous example), the TDU flows can carry additional control vector information for APPN resources. A new x'4E' control vector, named the Topology Resource Sequence Number Update control vector, can be included on topology updates for an APPN resource (either a node or a TG).

A sub-vector x'80' is defined within the x'4E' control vector, and two sub-vector x'80' instances can be included within in a single x'4E'. Each sub-vector x'80' captures information about an update of this resource, in particular, the name of the CP that generated the update. By capturing two instances of the sub-vector x'80', the CV x'4E' can provide awareness of the two most recent updates of the resource. This will typically be sufficient for identifying the two nodes engaged in the TDU War.

The new start option TDUDIAG allows you to control the frequency of when the CV x'4E' is added to topology update flows. You can specify one of three values for TDUDIAG. The value ALWAYS can be specified to force an CV x'4E' to be included every time the resource is updated by this node. The value NEVER can be used to disable this function such that a CV x'4E' is never included on any TDU flows. Finally, you can specify a numeric value for TDUDIAG for selective inclusion of the CV x'4E'. The numeric value represents a threshold of TDU updates for a given resource within a 24-hour window. If this threshold is reached, then the CV x'4E' is appended to any subsequent updates in that 24-hour period. The default, as shown on the slide, is a threshold value of 1000 updates in a 24-hour period.

Partner-specific handling of unknown topology vectors

- VTAM will now send unknown vectors to adjacent nodes that support them, even if one adjacent partner does not

- For example, CP2 wants to broadcast a TDU with the new CV x'4E' TDU War information

CP1 NN VM or VSE   **CP2 NN**   **CP5 NN**

Only CP1 does not get new topology information

CP3 NN   **CP4 NN**

Rest of APPN network is aware of the TDU updates

As APPN topology processing has progressed, additional topology control vectors, such as the CV x'4E' just discussed, have been defined. The control vectors added after the original APPN architecture are called, as a group, "unknown vectors". When the first such unknown vector was defined, the approach for processing them was to only forward unknown vectors to partner CP nodes if ALL partner CP nodes were capable to handling unknown vectors. If even one partner CP was not able to handle them, then the information was not sent to any partner nodes. This has led to situations where important topology information is not made known to the entire APPN network, resulting in degradation or improper route calculation.

Starting in z/OS V1R11, VTAM will now distribute unknown topology vectors on a node-by-node basis. TRS will include unknown control vectors on topology exchanges with all partners that can handle them properly, but will not include them when exchanging information with partners that can not. In the example on the slide, CP2 needs to broadcast new topology information, and in this situation is including the new CV x'4E' control vector for identifying nodes involved in a TDU War. Previously, since CP1 cannot handle unknown topology vectors, CP2 did not include the control vector on any of the updates. Now, however, only CP1 does not receive the new information. The other adjacent nodes do see the new control vector, which can be crucial in the event of a TDU War starting in the network.

"Wrong" model APPL definition can be used

- Customers use model APPL definitions to reduce definition requirements

- Sometimes, a different model APPL definition is used than is expected
    - Intended definition is not active
    - Overlap between model definitions

Which model best fits APPL1?

```
APPLMOD2 VBUILD TYPE=APPL
APPL*      APPL   AUTH=(PASS,ACQ)
APPL?*     APPL   AUTH=(PASS,ACQ)
APP?1      APPL   AUTH=(PASS,ACQ)
APP??      APPL   AUTH=(PASS,ACQ)
A?PL*      APPL   AUTH=(PASS,ACQ)
```

The Winner is .... APPL?* !!!

Model APPL definitions were created to reduce the need for creating and maintaining many APPL definitions. One model APPL definition can be used to generate thousands of dynamic definitions for applications. A model definition name contains one or more wildcard characters in it. The '?' wildcard character represents exactly one character. The '*' wildcard character represents 0 or more characters.

When an application issues OPEN ACB, VTAM first checks to see if there is an APPL statement defined that is an exact match. If not, VTAM will try to find the model APPL definition statement that is the best match for the name supplied by the application. The search for the best name is a character by character comparison, from left to right. An actual character match is considered to be a better match than a wildcard and the single character wildcard ('?') is considered to be a better match than the multi-character wildcard ('*'). When a match is found, a dynamic application RDTE is created with the model APPL definition statement.

Customers have had problems when their application used a different model than what they expected. This can cause the application to misbehave if the parameters on the found model APPL definition statement are different than what the application requires.

A wrong model might be used if the intended definition is not active. A second possibility is where one model was expected to be used, but another was used because it was a slightly better match. In this example, all of the model definitions are a potential match for APPL1, but only one is the "best match". Which model is best? If it is active, the best model in this example is APPL?*.

Enhance DISPLAY MODELS command for APPL models

- Added APPL keyword to DISPLAY MODELS command

- Display NET,MODELS,APPL to see which model APPL is best

```
D NET,MODELS,APPL=appl1
IST097I DISPLAY ACCEPTED
IST350I DISPLAY TYPE = MODELS
IST2302I MODEL APPL?* IS THE BEST ACTIVE MATCH FOR APPL1
IST314I END
```

- Can display situations when there is no matching model

```
D NET,MODELS,APPL=appl1
IST097I DISPLAY ACCEPTED
IST350I DISPLAY TYPE = MODELS
IST2303I THERE IS NO ACTIVE MODEL MATCH FOR APPL1
IST2304I APPL1 ALREADY EXISTS, TYPE = APPL
IST314I END
```

This release is adding a new APPL keyword to the existing DISPLAY MODELS command. This allows you to input an application name and find out what model APPL statement is the "best match" for an application name. The display also shows if the name is already in use so you can determine if there is a potential name conflict.

In the first example, the APPLMOD2 major node from the previous slide is active and no static definition for APPL1 exists. IST2302I is displayed and APPL?* is indicated as the best, active match.

In the second example, the APPLMOD2 major node is not active and there is a static definition for APPL1. IST2303I is displayed, which says that no active model was found that was a good match for APPL1. IST2304I is also displayed, which says there is a definition for APPL1 that already exists. In this case, it is an APPL. It is possible for it to be another definition type, such as a CDRSC. When you see the IST2304I message, you should see what type of resource is already using the name and determine if the name will present a conflict when the ACB is opened.

ACF/TAP provided as part of z/OS Communications Server

- ACF/TAP (Trace Analysis Program), previously delivered as part of NCP/SSP, provides good formatting and analysis of VTAM buffer traces

- VTAM will now provide ACF/TAP to allow customers to use ACF/TAP and debug their problems

  – No need to obtain SSP license just to use ACF/TAP

 – Same level of functionality provided
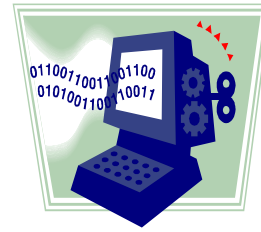
- New book added to z/OS Communication Server library:

  – z/OS V1R11 Communications Server: SNA ACF/TAP Trace Analysis Handbook (GC23-8588-00)

The ACF/TAP (Trace Analysis Program), previously delivered as part of NCP/SSP, provides good formatting and analysis of VTAM buffer traces, better than alternatives currently available with z/OS Communications Server alone. As users migrate away from NCP, they no longer need an SSP license either, but without the SSP license, users no longer have access to ACF/TAP.

To accommodate these customers, z/OS Communications Server will include ACF/TAP as part of the product, allowing customers to continue using ACF/TAP without an SSP license. The same level of functionality is being provided by z/OS Communications Server, and the ACF/TAP Trace Analysis Handbook has been added to the z/OS Communications Server library of manuals.

Data space VIT automatically collected with VTAM INOP DUMPs

- VTAM Internal Trace (VIT) table: 100-999 4K pages in ECSA
- Data space VIT table: 10 – 50 megabytes in ISTITDS1 data space
  - Number of VIT records preserved
    - 999-page ECSA VIT: 127,872 VIT records
    - 50-megabyte data space VIT: 1,638,400 VIT records
- VTAM INOP dumps
  - Taken automatically for certain inoperative conditions
  - Controlled by modifiable INOPDUMP start option
- VTAM will now include the data space VIT in INOP dumps
  - Reduces need for re-creates due to insufficient trace information
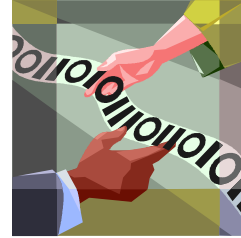
No externals to activate function

The VTAM Internal Trace (VIT) table is always recorded in ECSA. The SIZE operand on the TRACE,TYPE=VTAM start option controls the number of pages allocated for the VIT in ECSA, from 100 to 999 4K pages.

The DSPSIZE operand on the same start option controls the size of an optional, larger VIT table for which the ISTITDS1 data space is allocated. DSPSIZE is specified as an integer from 0 to 5. The number specified for DSPSIZE represents a multiple of 10 megabytes of data space storage. The data space VIT allows a much longer history of VIT entries to be preserved than the ECSA VIT alone. While 127,872 VIT records can be preserved in the largest ECSA VIT table, 1,638,400 VIT records can be preserved in the largest data space VIT.

If the data space VIT is active, VTAM ABEND dumps created by FRR and ESTAE routines automatically dump the VIT data space. This was not the case for VTAM INOP dumps however. VTAM INOP dumps are dumps taken automatically by VTAM when certain inoperative conditions occur. The modifiable INOPDUMP start option controls whether VTAM takes these dumps when the inoperative conditions occur. With only the ECSA VIT included, INOP dumps often did not have enough VIT history to allow problem determination in some cases.

Now with z/OS V1R11 Communications Server, when the data space VIT is active, VTAM will now include it in INOP dumps. This is not configurable, and there are no functional externals. The extra trace records should reduce the need for problem re-creates due to insufficient trace information.

Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

VTAM   z/OS

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

Other company, product, or service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2009. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.