

TCP/IP Directions for CICS Interconnectivity

Ian J Mitchell, IBM Distinguished Engineer
ianj_mitchell@uk.ibm.com

5th August 2010



SHARE in Boston

Objectives



As part of the multi-release IP interconnectivity (IPIC) initiative, CICS TS V4.1 now provides the facility to use TCP/IP instead of SNA for key base intercommunication functions, including 3270 based transaction routing, START and CANCEL commands. This is in addition to Distributed Program Link (DPL) introduced in CICS TS V3.2 that also provides for client requests via the CICS Transaction Gateway to invoke CICS programs using the Channels and Containers programming model and the ECI interface. This session will provide in-depth guidance on how to configure both your CICS and CICS Transaction Gateway systems to fully exploit the new IP interconnectivity (IPIC) support, and provides details on how the qualities of security, transactionality and connection management are best managed along with insight into future CICS connectivity strategy

- **Agenda**
 1. Overview of IPIC
 2. Release roadmap
 3. CICS configuration
 4. Security options
 5. Problem determination
 6. Performance
 7. CTG exploitation
 8. Migration

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

What is IPIC?

- **IPIC – IP InterCommunications Protocol**
 - Initiative to provide CICS communications support over TCP/IP as an alternative to that provided over ISC and MRO
- Customers require a TCP/IP alternative to SNA for CICS communications
 - Network convergence and simplification
 - SNA skills shortage
 - Cost of infrastructure
 - Take advantage of TCP/IP features on z/OS
- Multi-version delivery
 - No plan to remove existing SNA support
 - Aim is for migration of infrastructure without modification of CICS applications
 - Provide CICS with IP choice for most of the CICS coms-related programming model
- White paper 'CICS delivers IP interconnectivity' available at:
 - <http://www.ibm.com/software/hcp/cics/tserver/v32/library/index6.html>

Advantages of using TCP/IP

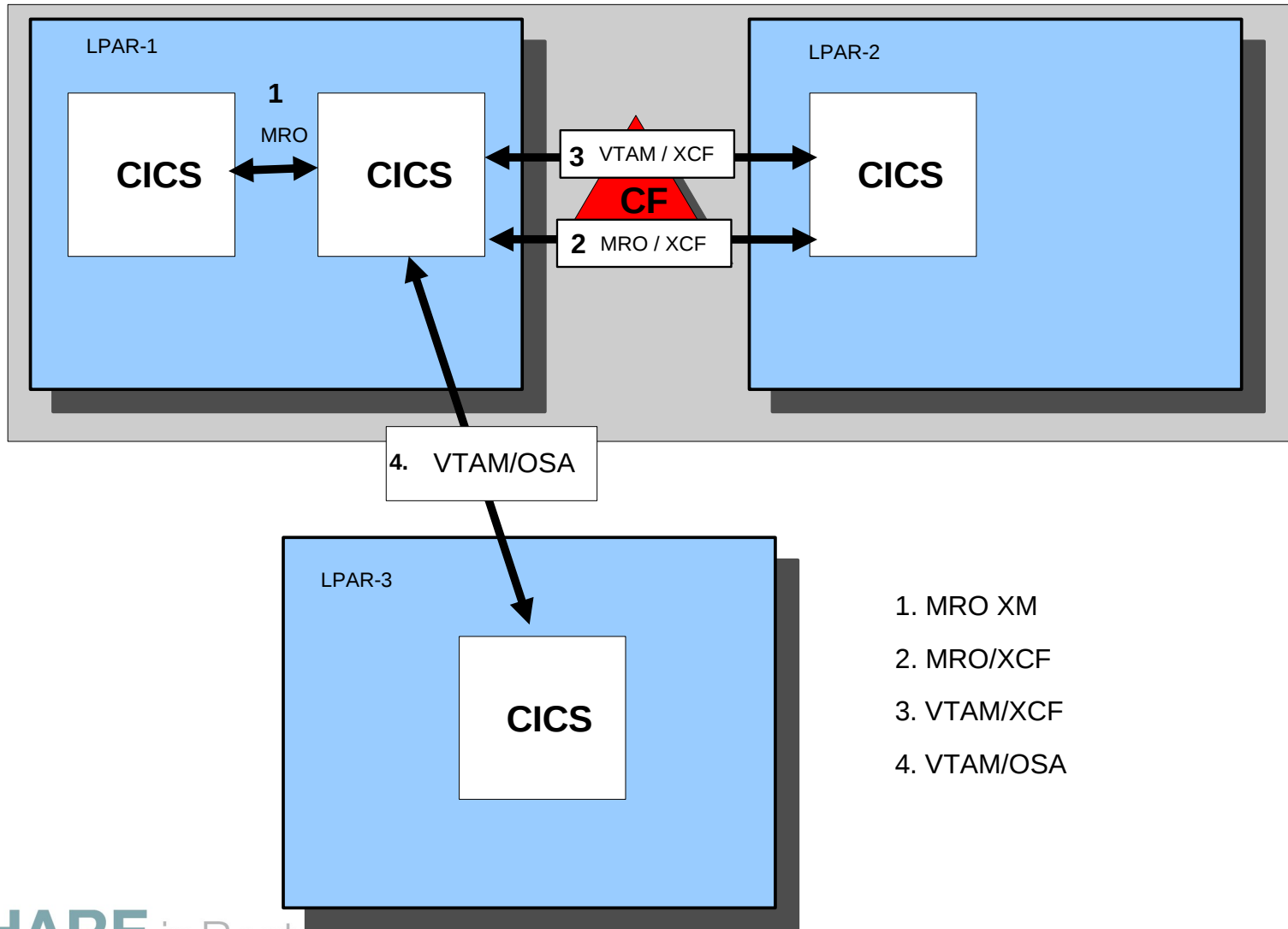
- Performance:
 - Exploitation of high bandwidth OSA connectivity (QDIO)
 - Lower total CPU usage
- Security
 - Standards based security (SSL and TLS)
 - System z hardware crypto support
 - Ongoing support of new security protocols
- Costs
 - Reduction in SNA skills required
 - Reduction in network hardware required
 - Improved problem determination facilities

IPIC functions and releases

Function	CICS TS 3.2	CICS TS 4.1	TXSeries 7.1
ECI requests	Yes	Yes	Yes (no 2pc)
3270 Txn routing	No	Yes	No
STARTs	No	Yes	No
DPL	Yes	Yes	Yes (no 2pc)
Function Shipping	No	No	No
CPSM SYSLINKs	No	Yes	n/a

- New in CICS TS V4.1
 - Asynchronous processing (STARTs) restricted to function shipping of START, START CHANNEL, and CANCEL commands
 - Supported with or without terminal
 - Transaction routing for 3270 terminals only
 - Traditional routing only – no enhanced routing ROUTABLE(YES)
 - No LU0 or LU6.2 session routing
 - TORs must be uniquely identified (no duplicate APPLIDs)
 - CICS TS V4.1 also provides:
 - CPSM definitions (SYSLINKs), IPv6 and DNS/hostname processing, Improved performance, Distributed Identities (requires z/OS 1.11)

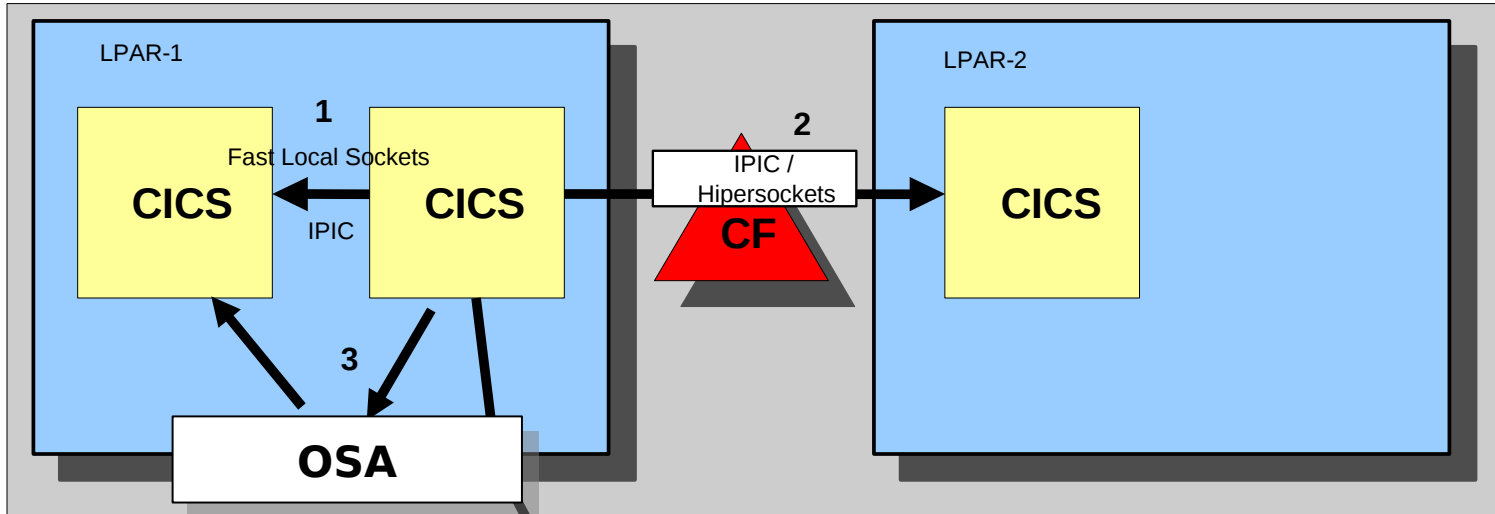
Existing connectivity (MRO / VTAM / ISC)



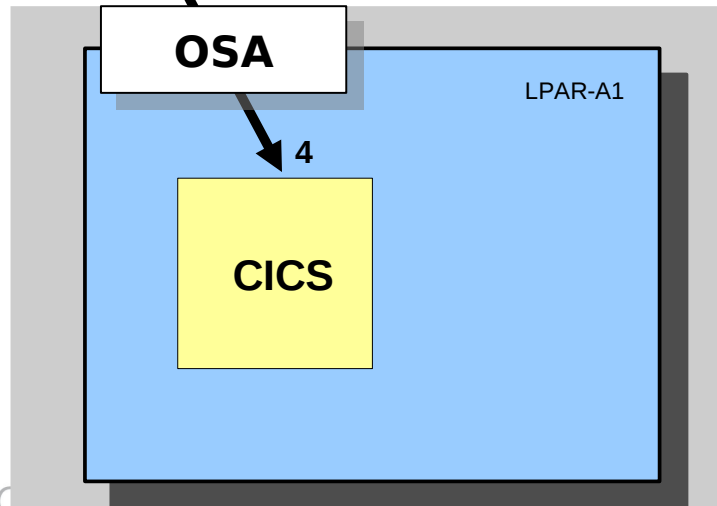
- 1. MRO XM
- 2. MRO/XCF
- 3. VTAM/XCF
- 4. VTAM/OSA

IP network options

Sysplex 1



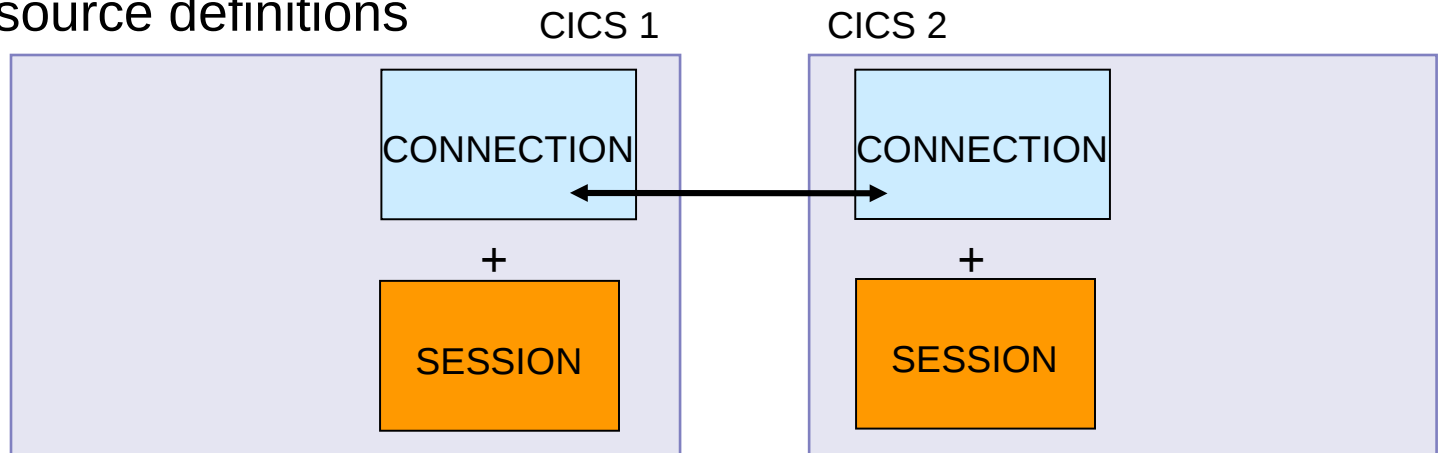
Sysplex 2



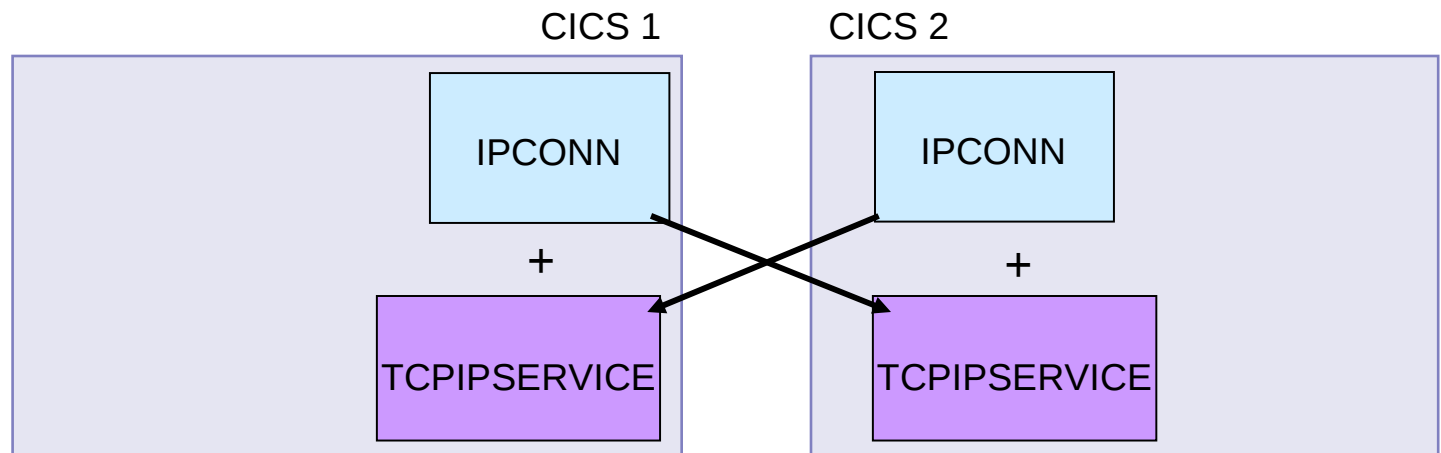
1. Fast local sockets
2. Hipersockets (same sysplex)
3. OSA (same sysplex)
4. OSA – Gigabit Ethernet

CICS SNA resource definitions

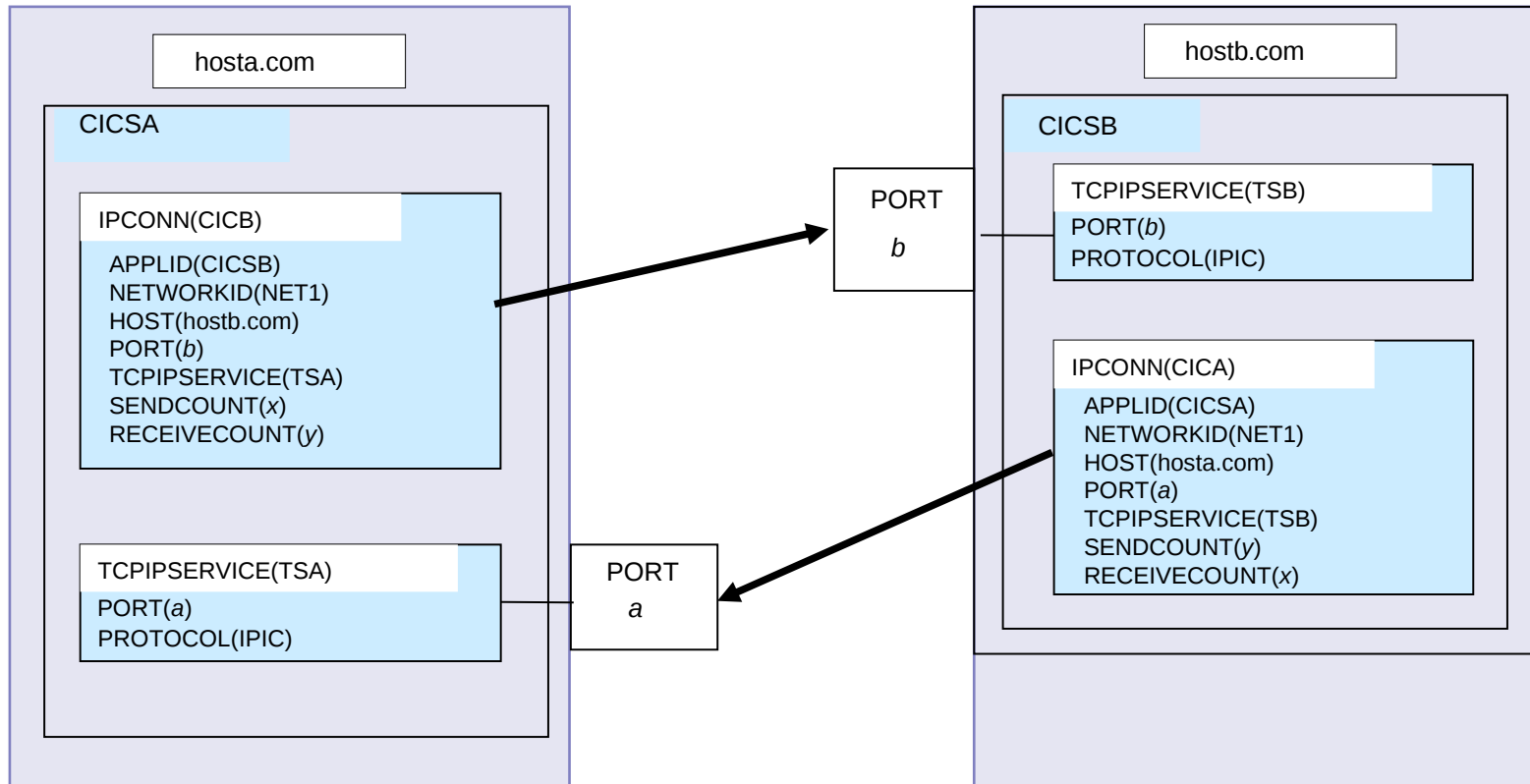
- SNA resource definitions



- IP interconnectivity resource definitions



IPIC resources example



IPCONNs

- Autoinstall supported via URM
 - Default is enabled
- Duplicate IPCONN names with SNA/MRO connections allowed
 - Provides for migration without changing CICS application SYSID parameters
- CICS to CICS requires two-way IPCONNS
 - One socket for each way
 - One IPCONN in each CICS region
- CICS TG to CICS uses a one-way IPCONN

IPCONN AUTOINSTALL

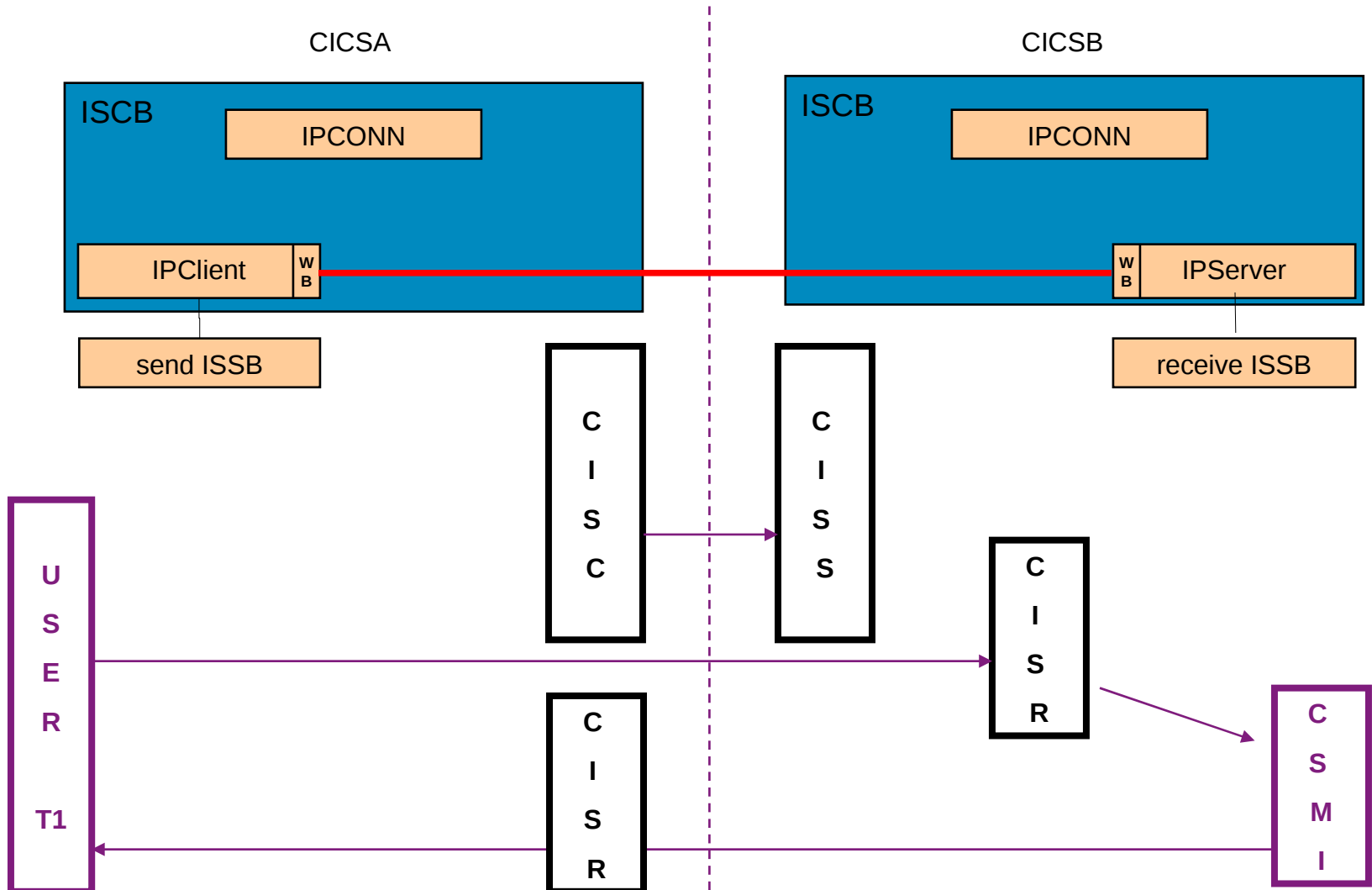
- Enabling AUTOINSTALL for IPCONN
 - TCPIP SERVICE must be defined with PROTOCOL (IPIC) and URM (DFHISAIP | program_name)
 - To disable autoinstall set **URM=NO**
 - Default is autoinstall is enabled

 - Invoked at INSTALL
 - Specifies name of installed IPCONN to be used as a template
 - May override APPLID, HOST and PORT

 - Invoked at DELETE
 - No action necessary

 - Assembler,C,COBOL,PL/I samples supplied in SDFHSAMP
 - DFHISAIP URM does not use a template
 - Refer to “DFHISCIP” in CTG infocentre for information on how to use an IPCONN template to set autoinstall USERAUTH security

DPL IPIC flows



IPIC transactions

- Service transactions
 - CSMI - EBCDIC mirror
 - CPMI – ASCII mirror
 - CRTE – Routing transaction

- CISB, CISC, CISD, CIST, CISS – Session management
- CISE - Error program
- CISR – IPIC request receiver
- CISX – XA recovery
- CISU – Recovery
- CISQ - Local queue processing
- CISM - Remote scheduler

Configuration

Enable TCP/IP

- TCP/IP
 - Allow stack access to CICS regions userid
 - EZB.STACKACCESS.sysname.tcpname
 - Reserve listening ports
 - PORT 6002 TCP CICS1 SAF Keyword
 - RACF protect ports (if required)
 - EZB.PORTACCESS.sysname.stackname.Keyword
- SIT parms:
 - MAXSOCKETS
 - SIT parameter controlling max sockets
 - Should be <= MAXFILEPROC in SYS1.PARMLIB
 - TCPIP=YES
 - Enables Web domain

Define TCPIPService

- **URM**
 - DFHISAIP (autoinstall) or NO to disable

- **Portnumber**
 - Local listening port

- **Protocol**
 - IPIC

- **TRANSACTION**
 - CISS

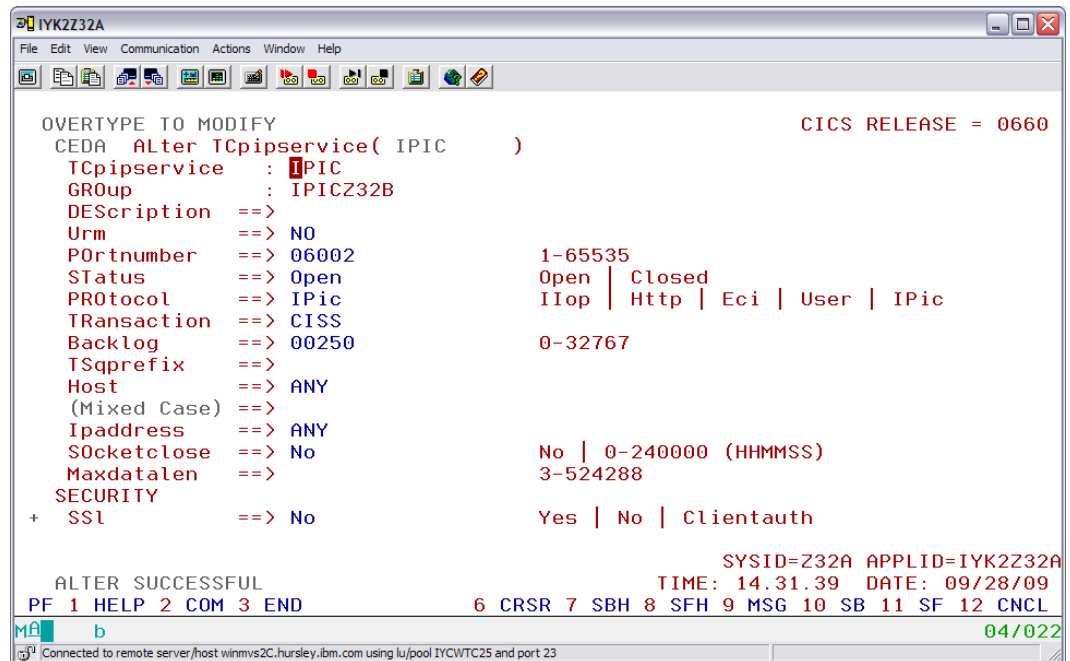
- **BACKLOG**
 - Set to max clients

- **HOSTNAME ***
 - ANY (binds to all stack IP addresses)

- **IPAddress**
 - Restricts bind to a specific IP address

- **Socketclose**
 - NO

- **SSL**
 - Enables SSL



```

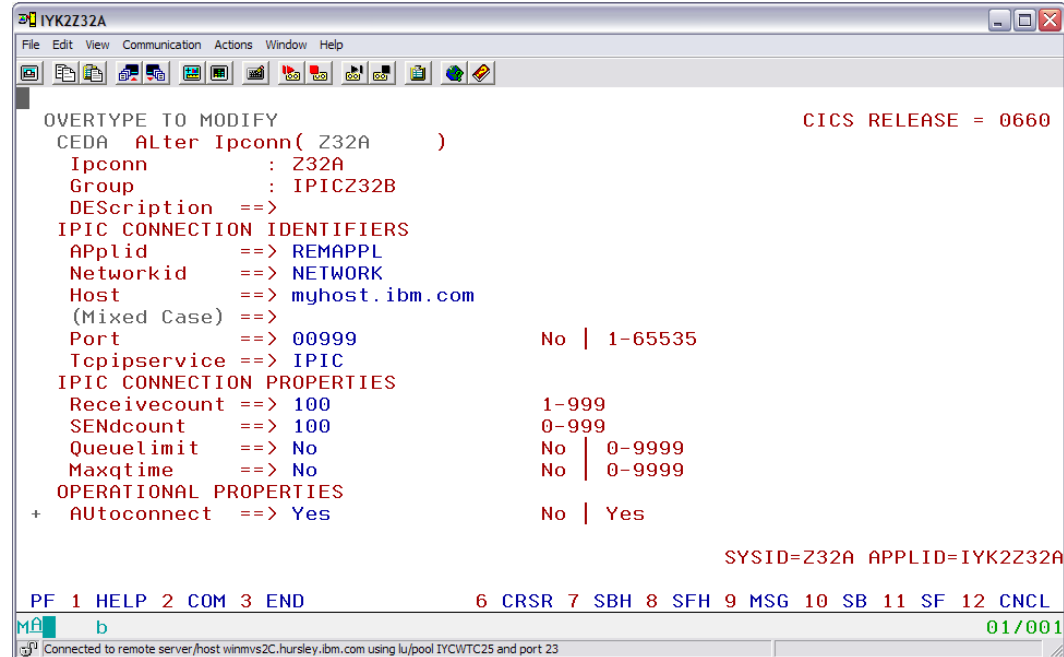
IYK2Z32A
File Edit View Communication Actions Window Help
OVERTYPE TO MODIFY                                CICS RELEASE = 0660
CEDA Alter TCPIPService( IPIC                      )
TCPIPService  : IPIC
GRoup        : IPICZ32B
DEscription  ==>
Urm          ==> NO
Portnumber   ==> 06002          1-65535
Status       ==> Open          Open | Closed
PRotocol     ==> IPic          Iiop | Http | Eci | User | IPic
TRansaction  ==> CISS
Backlog      ==> 00250          0-32767
TSqprefix    ==>
Host         ==> ANY
(Mixed Case) ==>
Ippaddress   ==> ANY
SOcketclose  ==> No           No | 0-240000 (HMMSS)
Maxdatalen   ==> 3-524288
SECURITY
+ SSL        ==> No           Yes | No | Clientauth

ALTER SUCCESSFUL                                SYSID=Z32A APPLID=IYK2Z32A
PF 1 HELP 2 COM 3 END                          TIME: 14.31.39 DATE: 09/28/09
6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
MÁ b                                           04/022
Connected to remote server/host winmvs2C.hursley.ibm.com using lu/pool IYCWTC25 and port 23
  
```

* New function in CICS TS V4.1:
 Hostname support, IPv6 support, and DNS lists

Define IPCONN

- **APPLID/NETWORKID**
 - Details of remote CICS or CTG
- **HOST**
 - TCP/IP hostname of remote system
- **PORT**
 - Listening port of remote CICS
 - Do **not** specify port for one way IPCONNs
- **TCPIP SERVICE**
 - Local TCPIPService
- **SEND COUNT/RECEIVE COUNT**
 - Number of Send/Receive sessions
 - Set to <= Maxtasks



```
IYK2Z32A
File Edit View Communication Actions Window Help
OVERTYPE TO MODIFY                                CICS RELEASE = 0660
CEDA Alter Ipconn( Z32A )
  Ipconn      : Z32A
  Group       : IPICZ32B
  Description  ==>
IPIC CONNECTION IDENTIFIERS
  APplid     ==> REMAPPL
  Networkid  ==> NETWORK
  Host       ==> myhost.ibm.com
  (Mixed Case) ==>
  Port       ==> 00999           No | 1-65535
  Tcpiptime  ==> IPIC
IPIC CONNECTION PROPERTIES
  Receivecount ==> 100         1-999
  SENDcount   ==> 100         0-999
  QueueLimit  ==> No          No | 0-9999
  Maxqtime    ==> No          No | 0-9999
OPERATIONAL PROPERTIES
+ Autoconnect ==> Yes         No | Yes
                                           SYSID=Z32A APPLID=IYK2Z32A
PF 1 HELP 2 COM 3 END                    6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
MÁ b                                     01/001
Connected to remote server/host winmvs2C.hursley.ibm.com using lu/pool IYCWTC25 and port 23
```

Validating configuration

- CEMT
 - INQ TCPIP
 - INQ TCPIPSERVICE
 - INQ IPCONN
- CPSM WUI:
 - Global TCP/IP statistics
 - Inbound and outbound sockets
- Msg logs
- Netstat

Validating configuration



CICS logs:

```
DFHS00107 09/28/2009 15:36:27 IYK2Z32A TCPIPSERVICE IPIC has been opened on port
06001 at IP address ANY
```

....

```
DFHIS2001 09/28/2009 16:33:01 IYK2Z32A Client web session 1 from applid IYK2Z32B
accepted for IPCONN Z32B.
```

```
DFHIS2001 09/28/2009 16:33:01 IYK2Z32A Client web session 2 from applid IYK2Z32B
accepted for IPCONN Z32B.
```

```
DFHIS2000 09/28/2009 16:33:01 IYK2Z32A Server web session 1 with applid IYK2Z32B
on host localhost(127.0.0.1), port 06002 acquired for IPCONN Z32B.
```

```
DFHIS2000 09/28/2009 16:33:01 IYK2Z32A Server web session 2 with applid IYK2Z32B
on host localhost(127.0.0.1), port 06002 acquired for IPCONN Z32B.
```

Validating IPCONNs - Netstat

NETSTAT command can be used to

1. Query servers listening on a specific port - NETSTAT ALLCON (PORT *nnnn*)
2. Query servers by job name - NETSTAT ALLCON (CLIENT *jobname*)
3. Query servers by IPaddress - NETSTAT ALLCON (IPADDR *ipaddress*)
4. Query servers by IPaddress - NETSTAT ALLCON (HOSTNAME *hostname*)

```

NETSTAT ALLCON (PORT 6002
MVS TCP/IP NETSTAT CS V1R9          TCPIP Name: TCPIP          14:54:10
User Id  Conn      Local Socket      Foreign Socket      State
-----  ----  -
CICSA    007C0DA6 127.0.0.1..44155  127.0.0.1..6002    Establish
CICSA    007C0DA2 127.0.0.1..44154  127.0.0.1..6002    Establish
CICSB    007C0DA7 127.0.0.1..6002   127.0.0.1..44155  Establish
CICSB    007C0DA3 127.0.0.1..6002   127.0.0.1..44154  Establish
CICSB    007BF03F 127.0.0.1..6002   0.0.0.0..0         Listen
    
```

2 socket established from CICSA to CICSB using localhost (127.0.0.1)

2 sockets established from CICSB to CICSA using localhost

CICS region listening on INADDRY_ANY 0.0.0.0

Note:

- CTG one-way IPCONNs can only be acquired when first used by CTG
- One-way IPCONNs can not be acquired for SPI/CEMT action

Socket usage

- CICS TS 4.1 to 4.1
 - 2 sockets for send, 2 for receive
 - Total 4 sockets per region
- CICS TS 3.2 to 4.1 or 3.2
 - 1 sockets for send, 1 for receive
 - Total 2 sockets per region
- CTG to CICS TS
 - 1 sockets for send
 - Total 1 sockets per region
- Query current socket usage via
 - CEMT I TCP
 - CEMT I TCPIPSERVICE
 - CEMT I IPCONN
 - TCP/IP statistics (wui)

 - D OMVS,L,PID=<pid>
 - Where *pid* is pid of SO TCB (use WUI)

IPIC Basic Checklist

- 1. Check CICS TCPIPService is Open
 - CEMT I TCPIPS
- 2. Check there is a listening socket
 - NETSTAT ALLCON (CLIENT *jobname*)
- 3. *Check IP route*
 - *Ping <hostname>*
- 4. *Acquire connection and check for CAPEX or autoinstall errors*
 - *CICS: CSMT*
 - *CTG: STDOUT/STDERR*

Using both IPIC and SNA?



- Will need an IPCONN and CONNECTION definition for both systems
- DPL will search for an IPCONN, then a CONNECTION
- IPCONN and CONNECTION with same name must have same APPLID, and vice versa
- If IPCONN is out of service, request will fail

User Exits



- XISQUE
 - IPIC equivalent of XZIQUE to control allocate queuing
 - Used to control the number of queued requests for sessions on IPIC connections

- XRSINDI
 - Invoked for install and discard of IPCONNs
 - UEIDIPCO resource type

- XISQLCL
 - Queuing of START NOCHECK command
 - New exit in 4.1

Security

Security



- Options to control connection access
 - TCPIP SERVICE
 - Bind to specific IP address (not ANY)

 - APPLIDs
 - Disable IPCONN autoinstall
 - Define specific IPCONNs with APPLID and IP address of client system

 - SAF controls (for same sysplex)
 - Use STACKACCESS and PORT statement to control access to stack
 - Use SERVAUTH and NET ACCESS zones to control which IP addresses can connect

 - SSL
 - Use SSL client authentication to control which IP clients can connect to the CICS system

USERAUTH on IPCONNs

- Options to control link security
 - LINK SECURITY
 - LINKAUTH parameter names the user representing the access – it can be:-
 - *SECUSER*
 - » *In which case the link user is specified via SECURITYNAME parameter*
 - *CERTUSER*
 - » *In which case the link user is mapped by RACF from the client's SSL certificate (dynamic)*
 - FLOWED USER
 - USERAUTH
 - *Local – Tasks run under default user ID*
 - *Identify – User ID is pre-authenticated – “Asserted Identity”*
 - *Verify – User ID and password required and authenticated*

SSL Support



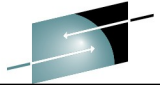
- Digital certificate-based authentication
- Asserts trust
 - “client” trusts “server”.
 - “server” trusts “client” (optional – client authentication)
- Encryption of the connection using strongest common cipher
- CICS allows cipher selection
 - in IPCONN (as “client”)
 - in TCPIPService (as “server”)
- Linkauth option to map client certificate to link user

Asserted Identity support

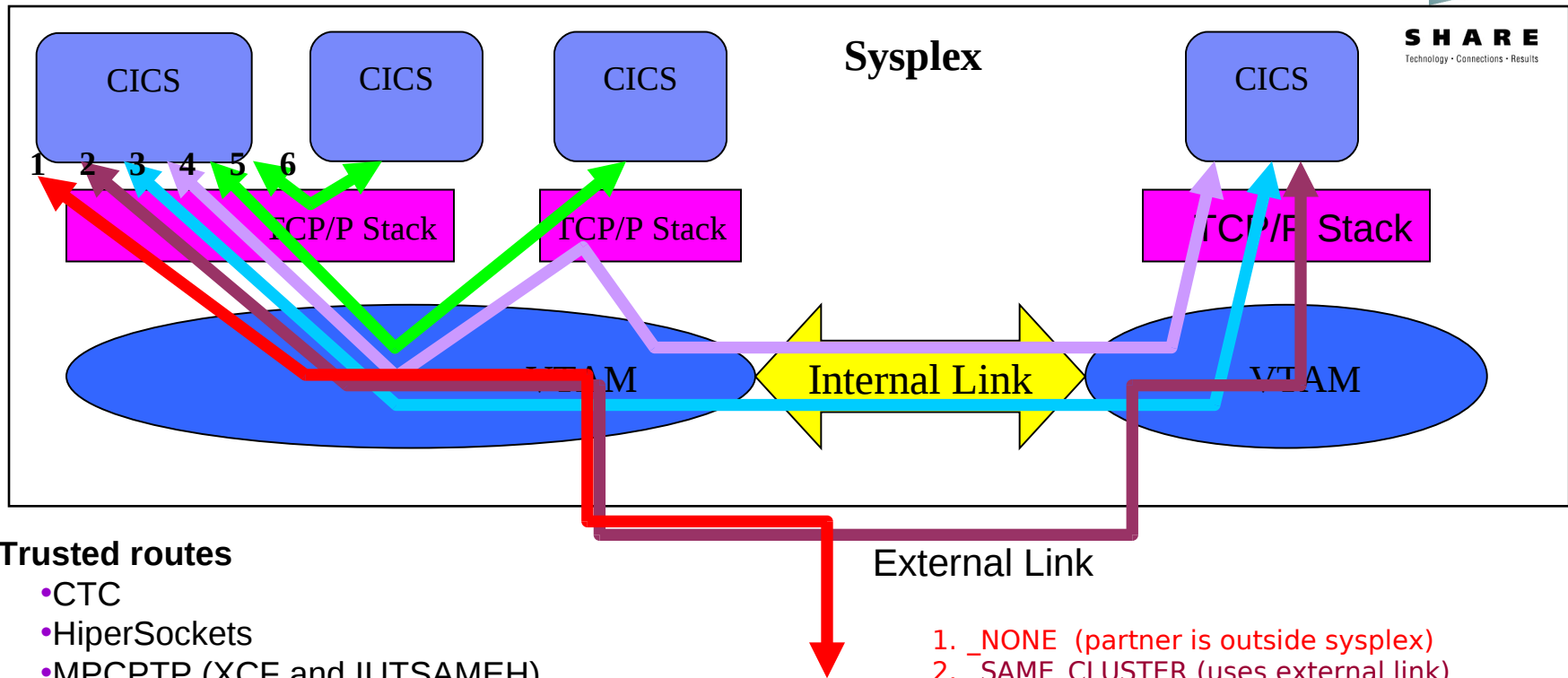


- Use of IPCONN USERAUTH (IDENTIFY) i.e. flowing of userid without a password now requires
 - Either SSL client authentication
TCPIPSERVICE SSL (CLIENTAUTH)
 - Or IPIC partner executes via *trusted IP route*
- Trusted IP routes:
 - Same sysplex or TCP/IP subplex
-
- Affects CICS to CICS and CICS TG to CICS requests

Asserted Identity - Trusted IP routes



SHARE
Technology • Connections • Results



- **Trusted routes**

- CTC
- HiperSockets
- MPCPTP (XCF and IUTSAMEH)
- Loopback (127.0.0.1)
- Same sysplex

- **Untrusted Routes**

- Partner is outside sysplex (1)
- Target address is a DVIPA
- Different TCP/IP subplex

1. `_NONE` (partner is outside sysplex)
2. `_SAME_CLUSTER` (uses external link)
3. `_SAME_CLUSTER, _INTERNAL *`
4. `_SAME_CLUSTER(uses indirect route)`
5. `_SAME_IMAGE, _SAME_CLUSTER, _INTERNAL *`
6. `_SAME_IMAGE, _SAME_CLUSTER, _INTERNAL *`

SO_CLUSTERCONNTYPE

- **Example-** Remote socket from Windows CTG (route 1)
 - Untrusted route
- **Netstat**
 - > TcpClusterConnFlag=x'81 ' (10000001)
- **CICS:**
 - > INQUIRE IPCONN CLIENTLOC('...0001')

Bit pattern	Netstat	Description	SO_CLUSTERCONNTYPE
0001	81	Remote socket	SO_CLUSTERCONNTYPE_NONE
0010	82	Different LPAR via OSA	SO_CLUSTERCONNTYPE_SAME_CLUSTER
0101	8A	Hipersocket	SO_CLUSTERCONNTYPE_INTERNAL SO_CLUSTERCONNTYPE_SAME_CLUSTER
1110	8E	Same LPAR Same TCP/IP stack Localhost	SO_CLUSTERCONNTYPE_SAME_IMAGE SO_CLUSTERCONNTYPE_INTERNAL SO_CLUSTERCONNTYPE_SAME_CLUSTER

RACF protection of TCP/IP networks

- SAF/RACF protects TCP/IP workloads just as it applies to all other types of workload
- SERVAUTH resource class used to define and protect TCP/IP resources
- General SERVAUTH profile format:
 - EZB.resource_category.system_name.jobname.resource_name
 - EZB designates that this is a TCP/IP profile
 - resource_category is capability area to be controlled e.g. TN3270, Stack Access, etc.
 - system_name is the name of the system - can be wildcarded (*)
 - jobname is the jobname associated with the resource access request - can be wildcarded (*)
 - optional resource_name - one or more qualifiers to indicate name of resource to be protected – can be wildcarded
- To protect one of the supported TCP/IP resources, you define a SERVAUTH profile with universal access(UACC) NONE and you then permit users to have READ access to the resources
- If the same sysplex but not using SSL client authentication recommendation is to use SERVAUTH class to authorize requests

Problem Determination

Monitoring options



- Debugging
 - Netstat and APPLDATA
 - CEDF

- Monitoring options:
 - Association data
 - APPLIDs (and CICS TG)
 - CICS Monitoring

CEDF

- CEDF
 - Can be used with DPLs, function shipped STARTs and ECI requests
 - CEDX <transaction>
 - Not supported with IPIC transaction routing
 - Abend AZTE
 - Not supported on connections (CEDF <sysid>)

APPLDATA

- **USS:** netstat -a -G *IYKZPM1*IPIC
- **MVS:** NETSTAT ALLCON (APPLD *IYKZPM1*IPIC

```

MVS TCP/IP NETSTAT CS V1R9          TCPIP Name: TCPIP          08:57:43
User Id  Conn      Local Socket          Foreign Socket        State
-----  -
CTGC32ZB 007C0DA7 127.0.0.1..6002      127.0.0.1..44155    Establish
Application Data: DFHIIYKZ2Z32BCISSIPIC  Z32A  IYK2Z32A
    
```



Task	Type
CIEP	ECI Inbound
CIRR	IIOB Inbound
CISS	IPIC Inbound
CISC	IPIC Outbound
CWXN	HTTP Inbound
CWXU	USER Inbound
Xxxx	HTTP Outbound
Xxxx	IIOB Outbound

Simple correlation mechanism for use with a socket and the application using the socket (CICS)

INQUIRE ASSOCIATION



CICS INQUIRE ASSOCIATION

- Retrieves information about how a task was started
- Built during task attach

CICS INQUIRE ASSOCIATION LIST

- A list of tasks with association data

```
>>-INQUIRE ASSOCIATION(data-value)-----+-----<<
                                     '-| options |-'
+-APPLDATA(data-area)-----+
+-APPLID(data-area)-----+
+-CLIENTIPADDR(data-area)-+
+-CLIENTPORT(data-area)----+
+-FACILNAME(data-area)-----+
+-FACILTYPE(cvda)-----+
+-INITUSERID(data-area)----+
+-IPCONN(data-area)-----+
+-IPFAMILY(cvda)-----+
+-LUNAME(data-area)-----+
+-MVSIMAGE(data-area)-----+
+-NETID(data-area)-----+
+-ODAPPLID(data-area)-----+
+-ODCLNTIPADDR(data-area)-+
+-ODCLNTPORT(data-area)----+
+-ODFACILNAME(data-area)--+
+-ODFACILTYPE(cvda)-----+
+-ODIPFAMILY(cvda)-----+
+-ODLUNAME(data-area)-----+
+-ODNETID(data-area)-----+
+-ODNETWORKID(data-area)--+
+-ODSTARTTIME(data-area)--+
+-ODTASKID(data-area)-----+
+-ODTRANSID(data-area)-----+
+-ODUSERID(data-area)-----+
+-PROGRAM(data-area)-----+
+-SERVERIPADDR(data-area)-+
+-SERVERPORT(data-area)----+
+-STARTTIME(data-area)-----+
+-TCPIPJOB(data-area)-----+
+-TCPIPSERVICE(data-area)-+
+-TCPIPZONE(data-area)-----+
+-TRNGRPID(data-area)-----+
+-TRANSACTION(data-area)--+
+-USERCORRDATA(data-area)-+
'-USERID(data-area)-----'
```

CPSM WUI - Task Association Data



- Activity – Tasks -> Task association data

Active tasks

[EYUVC1280I](#) 1 records collected at 2009/09/28 16:02:37.

CICS system name	IYK2Z32A
Task ID	0000122
Transaction	CSMI
User ID	CICSUSER

Associated task attributes

Originating network ID	GBIBMIYA
Originating application ID	CTGPW4
Originating start time	2009/09/28 16:00:21.3820
Originating task ID	00000000
Originating transaction ID	CSMI
Originating user ID	
Originating TCP/IP SERVICE name	
Originating portnumber	0
Originating client IP address	9.20.213.210
Originating client portnumber	4718
Originating transaction flags	0000000000000000
Originating facility name	
Originating user correlation data	

Name of system that originated the request

Transaction that initiated work at *Point of Origin*

IP address of *Point of Origin* (Java client when using CTG)

- Activity – Active Tasks -> TCP/IP usage data -> Number of IP facilities associated with task

IP facilities

[EYUVC12801](#) 1 records collected at 2009/09/28 16:10:12.

Context:
 Scope: Automatic refresh:
 IP facility token: =
 Associated task id: =

Record	CICS system name	IP facility token	Associated task id	IP connection id	IP facility type
1	IYK2Z32A	2BA32A60	0000122	CTGPW4	PRINCIPAL

Resource name: IPFACIL. View name: EYUSTAR

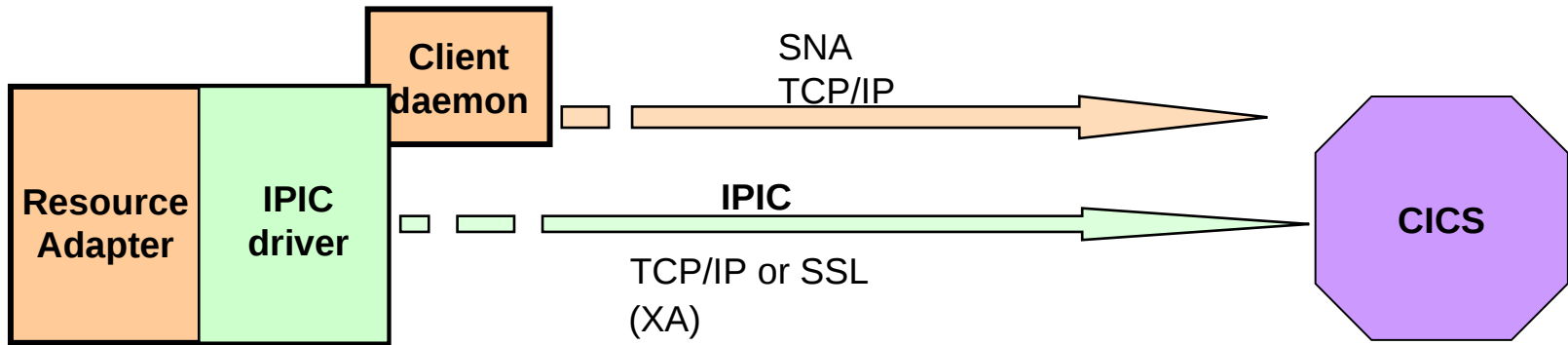
IPCONN used to initiate work
(contains link to TCPIPService)

Principal or Alternate IP
facility

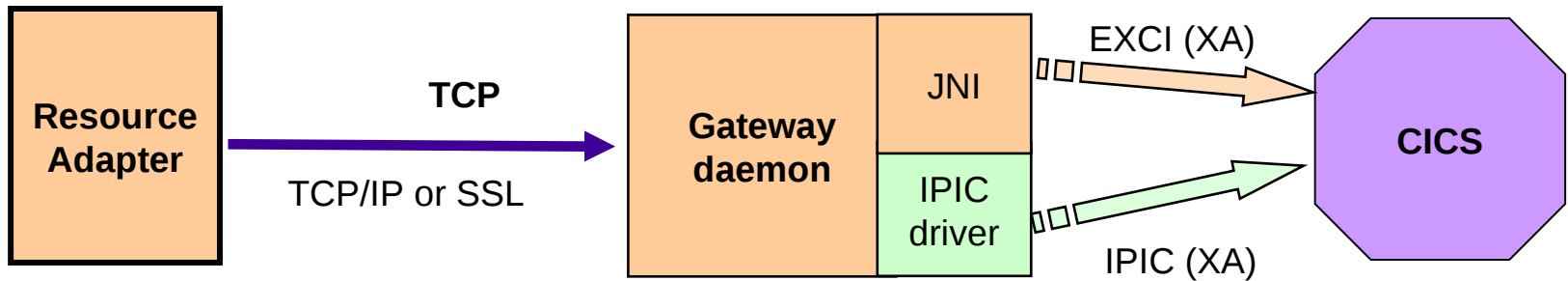
CICS TG Enhancements

CTG topologies

1. Local mode (2-tier)



2. Remote mode (3-tier)



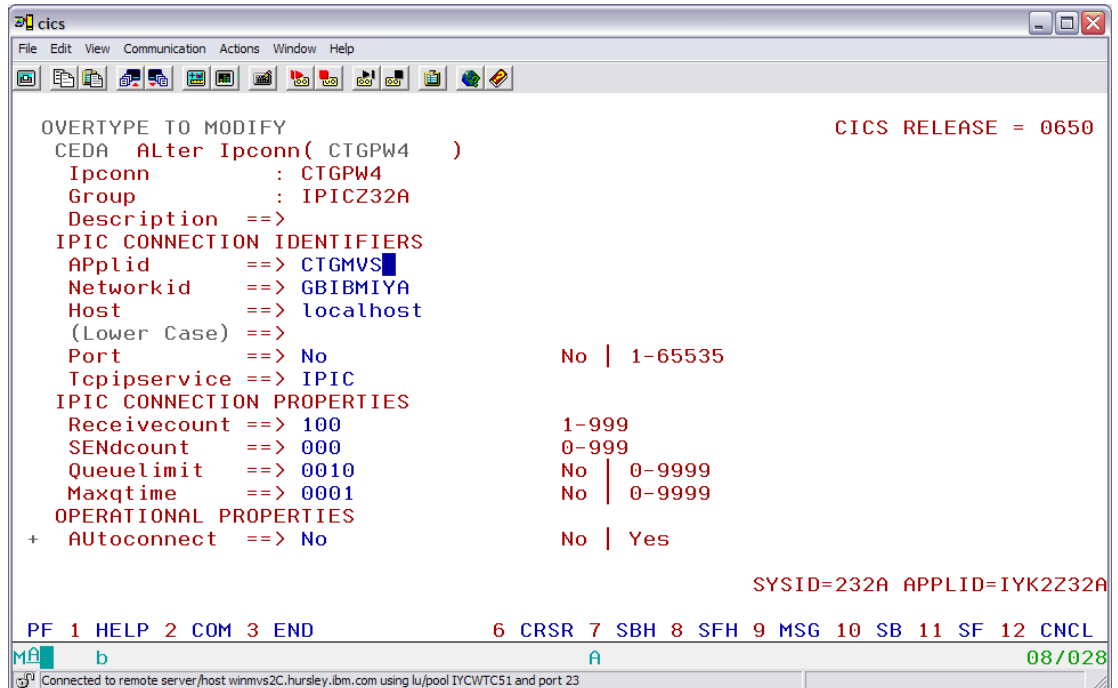
CTG IPIC Support



- Support for channels and containers
- Java/JCA - (CTG V7.1)
- ECIv2 Channels/Containers - (CTG V8)
- XA transaction (2pc) support
- SSL connections direct to CICS TS
- New monitoring options: Origin Data
- Fully zAAP enabled code path
- Asserted identity options
- Dual sockets support (CTG V8)

Define IPCONN for CTG

- **Applid:** Applid of CTG (optional)
- **NetworkID:** APPLIDQUALIFIER of CTG (optional)
- **Host:** TCP/IP hostname for remote CTG(optional) use localhost if on same TCP/IP stack
- **Port:** NONE (Do not specify port for CTG one way IPCONN)
- **TcpipService:** CICS TCPIPService listening for requests
- **Receivecount:** Number of receive sessions to service parallel requests
set to <= MAXTASKS
- **SendCount:** N/A - Set to 000 for one-way IPCONNs
- **Queuelimit:** Number of requests to queue waiting for a receive session
- **Autoconnect:** No



```

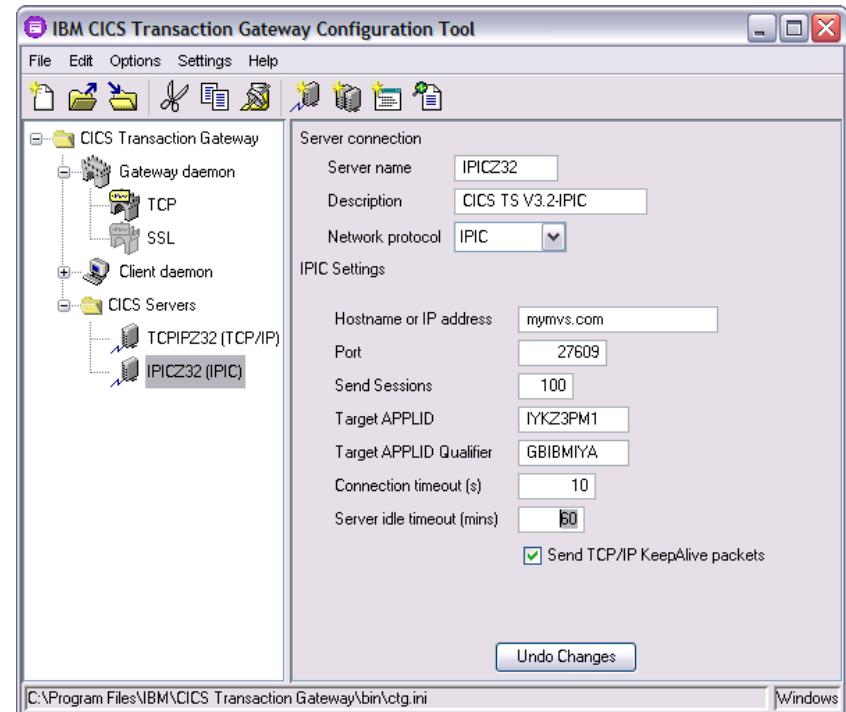
cics
File Edit View Communication Actions Window Help
OVERTYPE TO MODIFY                                CICS RELEASE = 0650
CEDA ALter Ipconn( CTGPW4 )
  Ipconn      : CTGPW4
  Group       : IPICZ32A
  Description ==>
IPIC CONNECTION IDENTIFIERS
  APplid      ==> CTGMVS
  Networkid   ==> GBIBMIYA
  Host        ==> localhost
  (Lower Case) ==>
  Port        ==> No                No | 1-65535
  Tcpiervice ==> IPIC
IPIC CONNECTION PROPERTIES
  Receivecount ==> 100                1-999
  SENDcount    ==> 000                0-999
  Queuelimit   ==> 0010              No | 0-9999
  Maxqtime     ==> 0001              No | 0-9999
OPERATIONAL PROPERTIES
+ Autoconnect ==> No                No | Yes

SYSID=232A APPLID=IYK2Z32A
PF 1 HELP 2 COM 3 END                6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
MA b                                  A 08/028
Connected to remote server/host winmvs2C.hursley.ibm.com using lu/pool IYCWTC51 and port 23
```

IPIC definitions – Gateway daemon

- Ctg.ini file:
- SECTION PRODUCT
 - APPLID=CTGMVS
 - APPLIDQUALIFIER=GBIBMIYA (mandatory to match hard coded IPCONNns)
 - DEFAULTSERVER=IPICZ32
- ENDSECTION

- SECTION IPICSERVER=IPICZ32
 - DESCRIPTION=CICS TS V3.2 IPIC server
- TCPKEEPALIVE=Y
- SRVIDLETIMEOUT=60
- SENDSESSIONS=100
- CONNECTTIMEOUT=10
- HOSTNAME=localhost
- PORT=27609
- CICSAPPLID=IYKZ3PM1
- CICSAPPLIDQUALIFIER=GBIBMIYA
- ENDSECTION



IPIC – local mode



- Servers are not defined in ctg.ini file or returned by ListSystems
- URL format is used
- CICS applid is optional, but will be used to verify connection

1. Protocol://hostname:port
2. Protocol://hostname:port#CICSAPPLID
3. Protocol://hostname:port#CICSAPPLIDQUALIFIER.CICSAPPLID

- Example

```
java com.ibm.ctg.samples.eci.EciB2 jgate=local: server=tcp://mymvs.com:27609 prog0=EC01 commarea=xx commarealength=17
```

<u>TranName</u>		<u>TranName</u>	<u>false</u>
<u>ConnectionURL</u>	<u>local:</u>	<u>ConnectionURL</u>	<u>false</u>
<u>ServerName</u>	<u>tcp://mymvs.com:27609#GBIBMIYA.IYKZ3PM1</u>	<u>ServerName</u>	<u>false</u>
<u>ClientSecurity</u>		<u>ClientSecurity</u>	<u>false</u>
<u>KeyRingPassword</u>		<u>KeyRingPassword</u>	<u>false</u>

CICS and CTG logs



- CICS displays

```
DFHS00107 09/28/2009 15:36:27 IYK2Z32A TCPIP SERVICE IPIC
has been opened on port 06001 at IP address ANY
```

.....

```
DFHIS2001 09/28/2009 16:30:28 IYK2Z32A Client web session
1 from applid CTGPW4 accepted for IPCONN CTGPW4
```

- CICS TG displays

```
09/28/09 16:30:28:261 [0] CTG8429I Established new IPIC
connection to CICS server IPICZ32A with: negotiated
session limit=100, CICSAPPLID=IYK2Z32A
CICSAPPLIDQUALIFIER=GBIBMIYA, HOSTNAME=localhost,
PORT=6001
```

Migration

Migration utility



- Migration utility for converting APPC and MRO connections to IPIC
 - DFH0IPCC sample program for use with DFHCSDUP system definition utility
 - Provide list of APPLIDs with corresponding hostnames and port numbers
 - Creates IPCONNs and TCPIPServiceS for connections

```
*****
APPLID. |NET-ID. |PORT.|HOST.
*****
.DEFAULT|LOCALNET|      |TCPIPServiceS=TCPSERV1
APPL1A  |          |9876 |my.local.hostname
OTHERCIC|OTHERNET|12345|this.host.has.a.very.long.name.which.is.going.to.require*
      |      |      |e.a.continuation.record
* Comments such as this are entirely free-form other than the * in column 1
CICSXYZ |          |9875 |10.2.156.221
```


SNA/IPIC Command equivalents



SNA	IPIC	Notes
CONN install warm start	IPCONN install warm start	Acquired if autoconnect(yes)
SET VTAM open SET CONN acq	SET TCPIP open SET TCPIPS open SET IPCONN acq	All conns with autoconnect(yes) acquired Refing IPCONNs with autoconnect(yes) acquired
SET VTAM closed SET CONN rel	SET TCPIPS (*) closed SET TCPIPS closed SET IPCONN rel	new allocates rejected, queued allocates and work in progress allowed
SET VTAM immclose SET VTAM forceclose	SET TCP closed SET TCP immclosed SET TCPIPS immclosed	new allocates rejected, queued allocates cancelled, work in progress abended
SET MODENAME CLOSED SET MODENAME AVAIL(0)	No equivalent	no concept of modegroup or cnos

CICS TS V4.1 – IP Performance

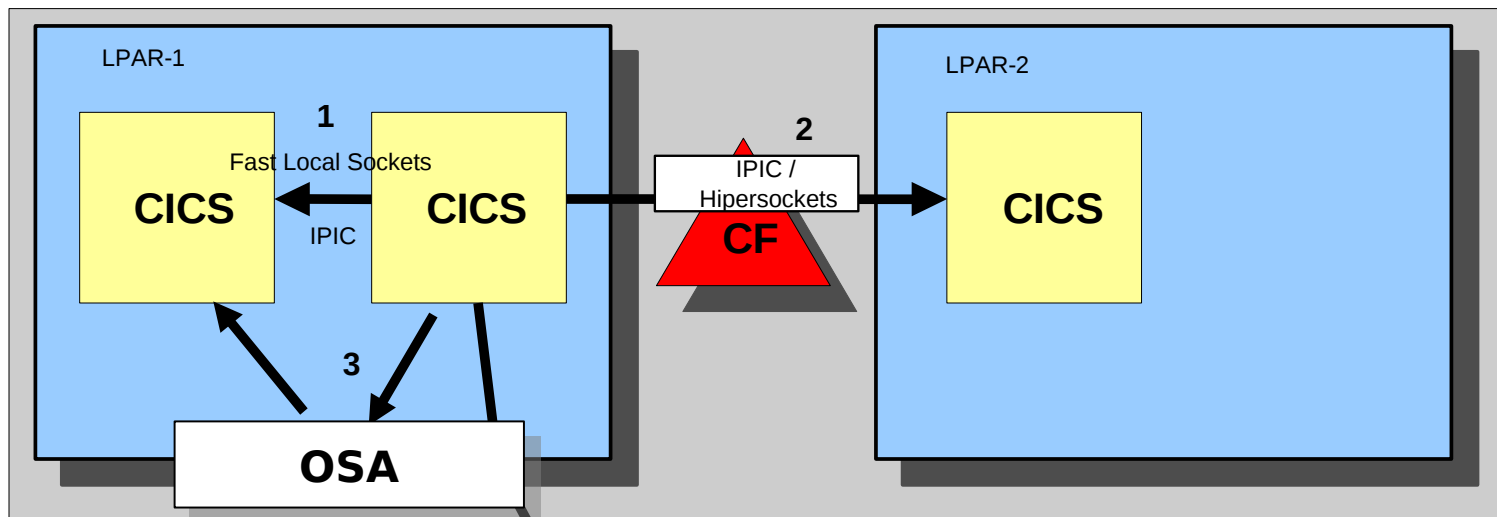


- CICS:
 - Improved pathlength for IPIC syncpoint flows
 - Multiple sockets
 - Improves scalability and response times under high load
 - Requires CTG V8.0 if using ECI over IPIC
 - Dynamic right sizing
 - Dynamic setting of 64K receive buffers to enable TCP/IP window scaling.
 - Enables the TCP/IP stack to dynamically optimise the window size
 - Prevents low data transmissions due to bursts of activity

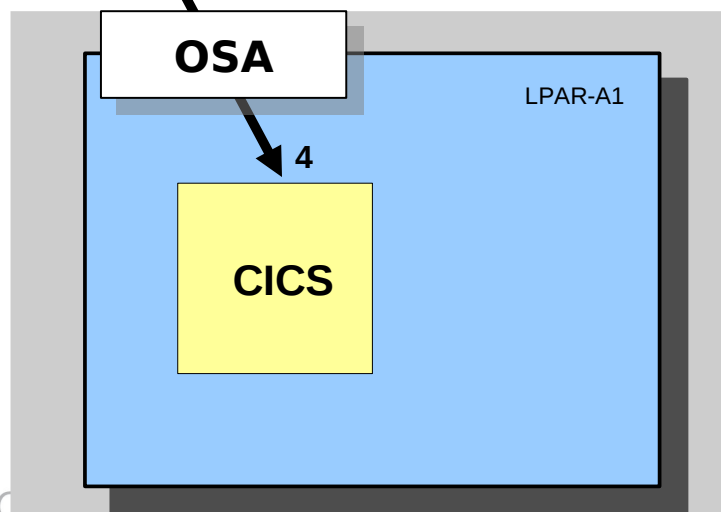
- z/OS TCP/IP networking improvements
 - 10-Gigabit ethernet
 - OSA-Express and QDIO
 - OSA-Express3
 - Optimized latency mode (OLM)
 - *Significant response time improvements for transactional workloads*
 - *V1.11 APR enablement PK90205 and OA29634*
 - zIIP-Assisted HiperSockets
 - z/OS V1.10

Benchmarking network options

Sysplex 1



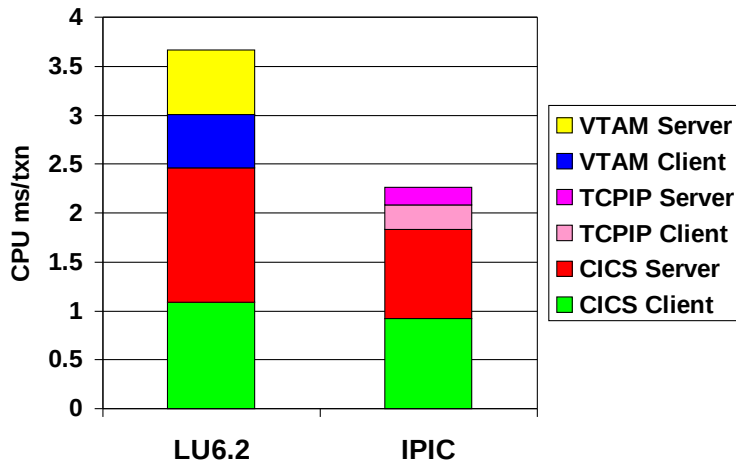
Sysplex 2



1. Fast local sockets
2. Hipersockets (same sysplex)
3. OSA (same sysplex)
4. OSA – Gigabit Ethernet

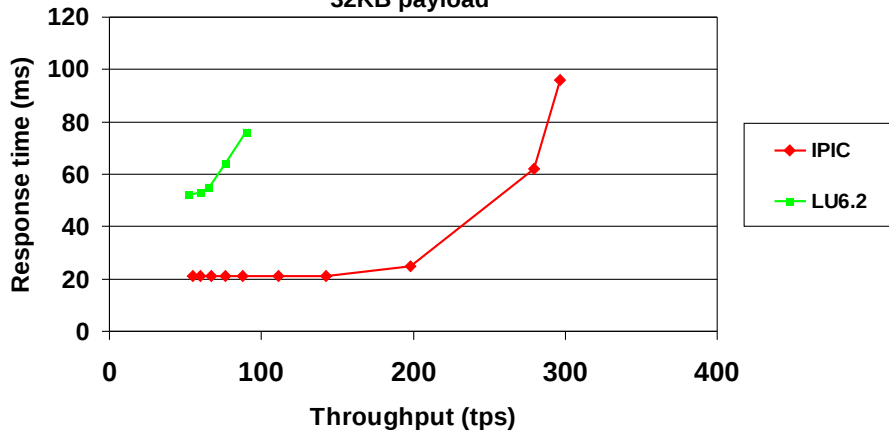
IPIC Performance vs SNA

CICS TS 4.1 DPL CPU usage, OSA via Gigabit Ethernet
32KB payload



- IPIC CICS pathlength comparable or less than LU6.2 pathlength
- As payload increases IPIC pathlength scales better than LU6.2
- CICS TS 4.1 DPL pathlength reduced by ~ 14% vs. CICS TS 3.2

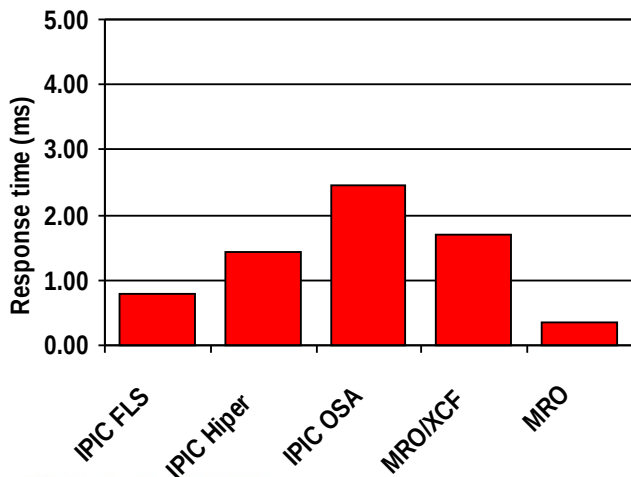
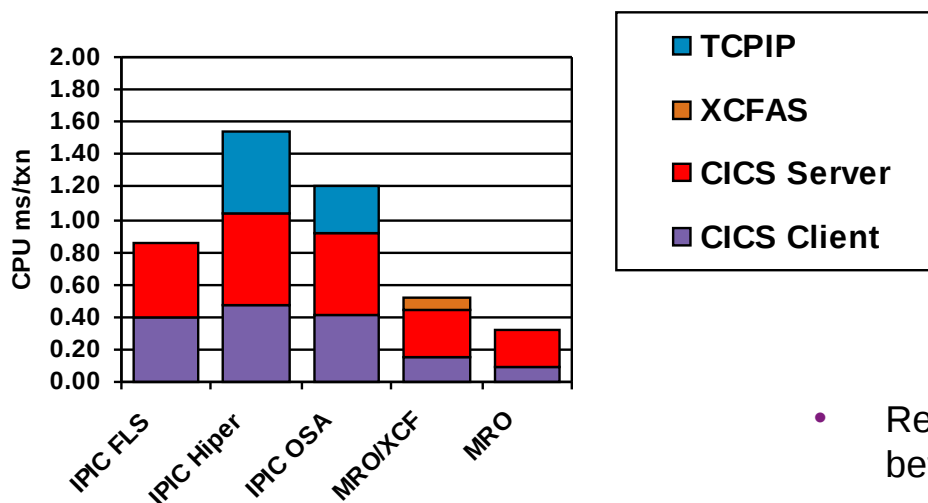
CICS TS 4.1 Response times, OSA via Gigabit Ethernet
32KB payload



- CICS to CICS IPIC response time better than LU6.2
- IPIC max throughput significantly better than LU6.2
- IPIC data throughput close to 75% of Gigabit ethernet
- Hipersockets gives very low latency for connections between LPARs
- IP via OSA uses less CPU than via Hipersockets

IP Routes vs. MRO

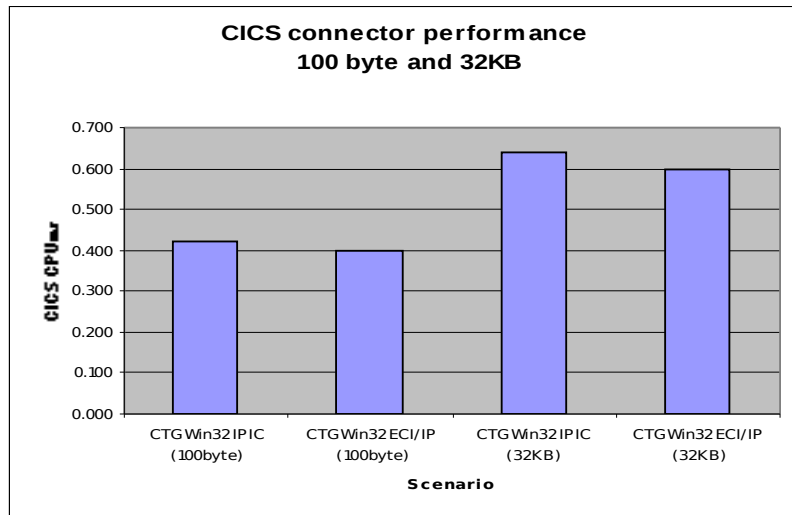
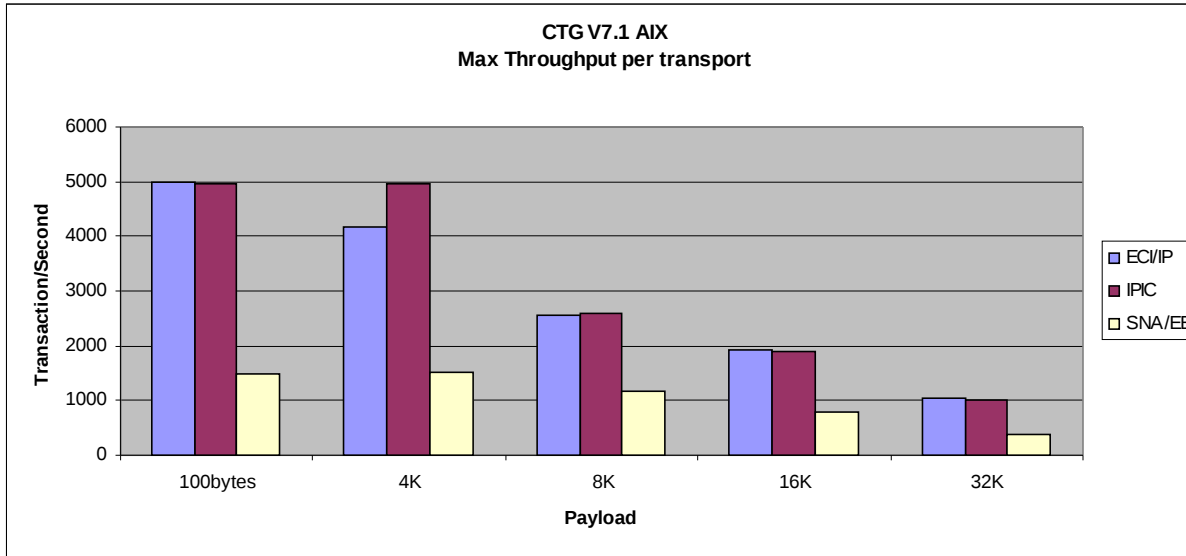
CICS TS 3.2 DPL – Total CPU Usage IPIC vs. MRO Transports, 32KB data/link



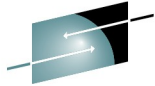
- Response time of IPIC/Hipersockets is equal or better than MRO/XCF
- IPIC via OSA uses less CPU than via Hipersockets
- Fast local sockets is most efficient and fastest IPIC route, but MRO has lowest CPU cost
- IP over OSA provides lower CPU cost than Hipersockets route for inter-LPAR comms

IPIC Performance

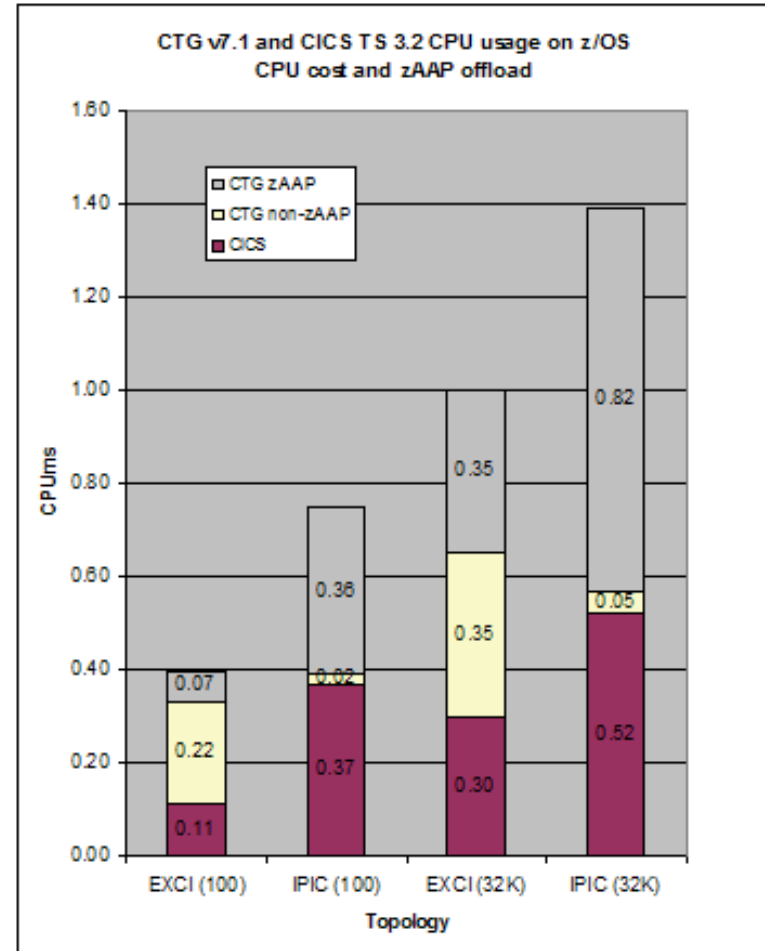
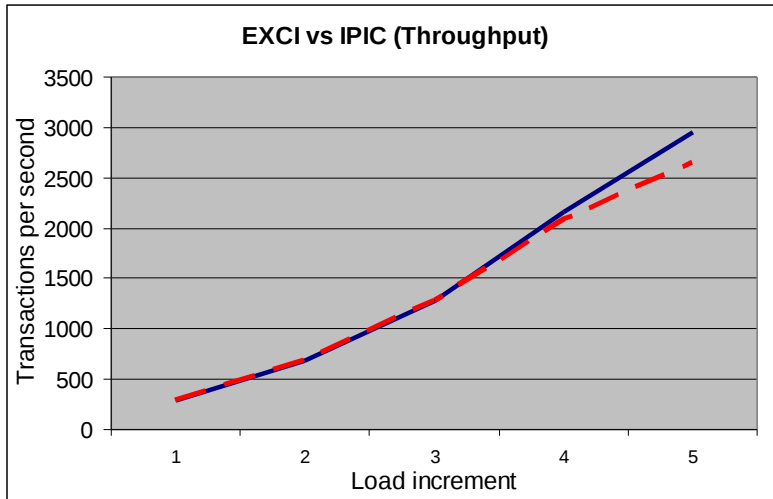
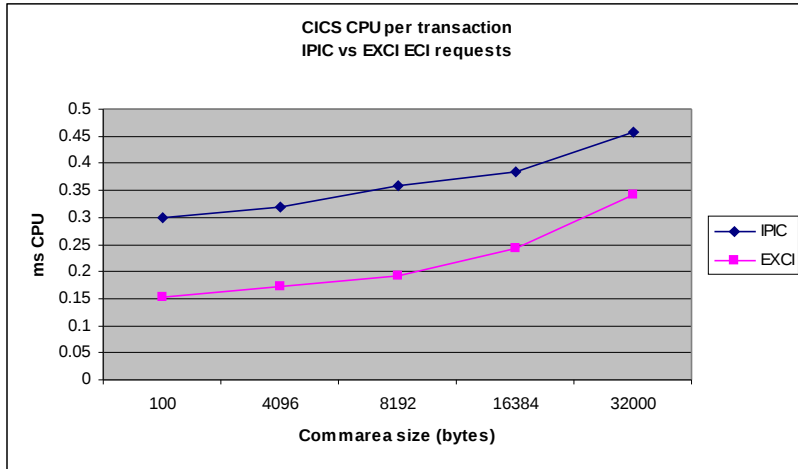
- CICS TG Multiplatform and ECI/IP



IPIC Performance - CICS TG z/OS and EXCI



SHARE
Technology • Connections • Results



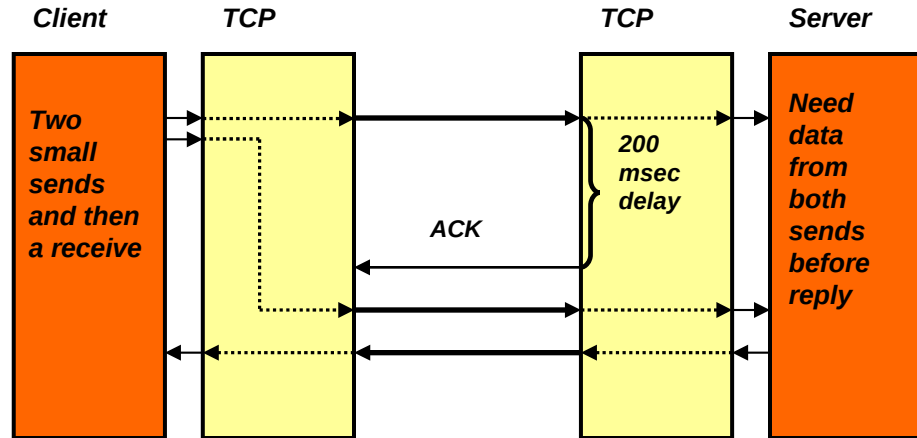
Analysing IP waits

- TCP/IP I/O Waits
 - CICS Monitoring data
 - *SUSPTIME*
 - *SOIOWTT*
 - If SOIOWTT wait time is high% of task wait time then delays in writing to socket
- TCP/IP
 - Netstat /A
 - Round-trip information: Smooth trip time:ms
 - *Time for data to be acknowledged*
 - High round trip time (200ms+) suggest delays in network or tcp/ip acknowledgements
- Send/receive buffers
 - Use 64K send/recv buffers if using socket connections into z/OS
 - TCPSENDBFRSIZE 65536
 - TCPRCVBFRSIZE 65536
- Ensure APAR PM19483 is applied if using 64K send/recv buffers with local sockets with V1R11 or earlier

z/OS V1R11 - Dynamic TCP NODELAY tuning

- **Nagle** (on send side)
 - Data from a small send() cannot be put on the wire if there is outstanding un-acknowledged data
 - Applications can disable Nagle by setting the TCP_NODELAY sockets options
 - CICS TS 3.2 disables Nagle for key protocols

- **Delayed ACK** (on receive side)
 - TCP generally ACKs every 2nd segment
 - TCP generally waits 200 msec before sending a stand-alone ACK if no 2nd segment arrives



New transactional applications often encounter severe performance problems due to this behavior

- Most application programmers don't know about Nagle
- Very often seen with CICS Sockets applications
- z/OS V1R11 Communications Server transparently relaxes the requirements of Nagle for the initial exchanges of data between two applications

✓ Improved performance for older CICS release that do not disable Nagle

Summary



- IP Advantages -
 - Increased QoS over CICS APPC connections
 - Reduced cost of IP networking
 - Wider range of security options
 - Ability to exploit networking IP optimizations
- Multi-release delivery
 - Allows staged migration away from SNA networks
 - Without impacting business applications
- For more details on CICS IP interconnectivity strategy refer to Whitepaper: “CICS delivers IP interconnectivity”
 - <http://www.ibm.com/software/htp/cics/tserver/v32/library/#wpapers>

Requirements

- CICS TS V4.1
 - PK91639 for XA

- CICS TS V3.2
 - APARs:
PK49017, PK49015, PK49116, PK49490, PK49021,
PK51587, PK53783
 - PK55716, PK57726, PK65134, PK91639 Required for XA
 - PK55494, PK55495 for asserted identity control

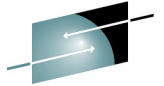
- CTG V7.1.0.2 (or later) for DPL over IPIC

Slide Title (Type Size=28) (no more than two lines)

- First Major Topic (Type Size=24)
 - Subtopic One (Type Size=22)
 - Subtopic Two (Type Size=22)
 - Sub-subtopic (Type Size=20)
- Second Major Topic (Type Size=24)
- Third Major Topic (Type Size=24)
- Fourth Major Topic (Type Size=24)

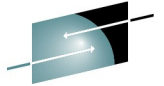
Two-Column Slide (Type Size=28)

- Topic A (Type Size=24)
 - Subtopic 1 (Type Size=22)
 - Subtopic 2 (Type Size=22)
 - Subtopic 3 (Type Size=22)
 - Subtopic 4 (Type Size=22)
- Topic B (Type Size=24)
- Topic C (Type Size=24)
 - Subtopic 1 (Type Size=22)
 - Subtopic 2 (Type Size=22)
 - Subtopic 3 (Type Size=22)
 - Sub-subtopic 1 (Type Size=20)
 - Sub-subtopic 2 (Type Size=20)
- Topic D (Type Size=20)



SHARE
Technology • Connections • Results

Slide with Table



SHARE
Technology • Connections • Results

Slide with Text & Graphic