CICS and IPIC Tutorial

Leigh Compton
IBM Corporation

Wednesday, 10 August 2011
Session 09610

Abstract

- IP interconnectivity (IPIC) is a newly introduced type of CICS intercommunication link that enables you to implement CICS-to-CICS communications using TCP/IP. IPIC supports the full capabilities of IP networking including IPv4, IPv6, SSL, and TLS. Once the IPIC connections are configured, CICS regions may transmit DPL, transaction routing, and function shipping requests over the IP network just as has been done over MRO and ISC connections. This tutorial will describe how IPIC connectivity works, how to create IPIC connections between CICS regions, and how to migrate existing MRO and ISC connections to IPIC

Agenda

- Introduction to IPIC
- Resource definitions
- Autoinstall
- Scenarios
  - CICS-to-CICS
  - CICS Transaction Gateway-to-CICS
- Migration planning

Introduction

- CICS TS V3.2 and above support three styles of intercommunication and transports
  - MRO, using cross memory (XM) or cross-system coupling facility (XCF)
  - ISC, using SNA/VTAM
  - IPIC, using TCP/IP
- IPIC initially targeted as alternative to SNA

IPIC staged delivery

- CICS TS V3.2
  - Distributed Program Link (DPL)
- CICS TS V4.1
  - 3270 transaction routing
  - Asynchronous processing (START)
  - ID propagation
- CICS TS V4.2
  - Function shipping
    - File control
    - Transient data
    - Temporary storage
- CICS TG V7.1 and above
  - ECI (DPL)
- TXSeries V7.2 and above
  - DPL

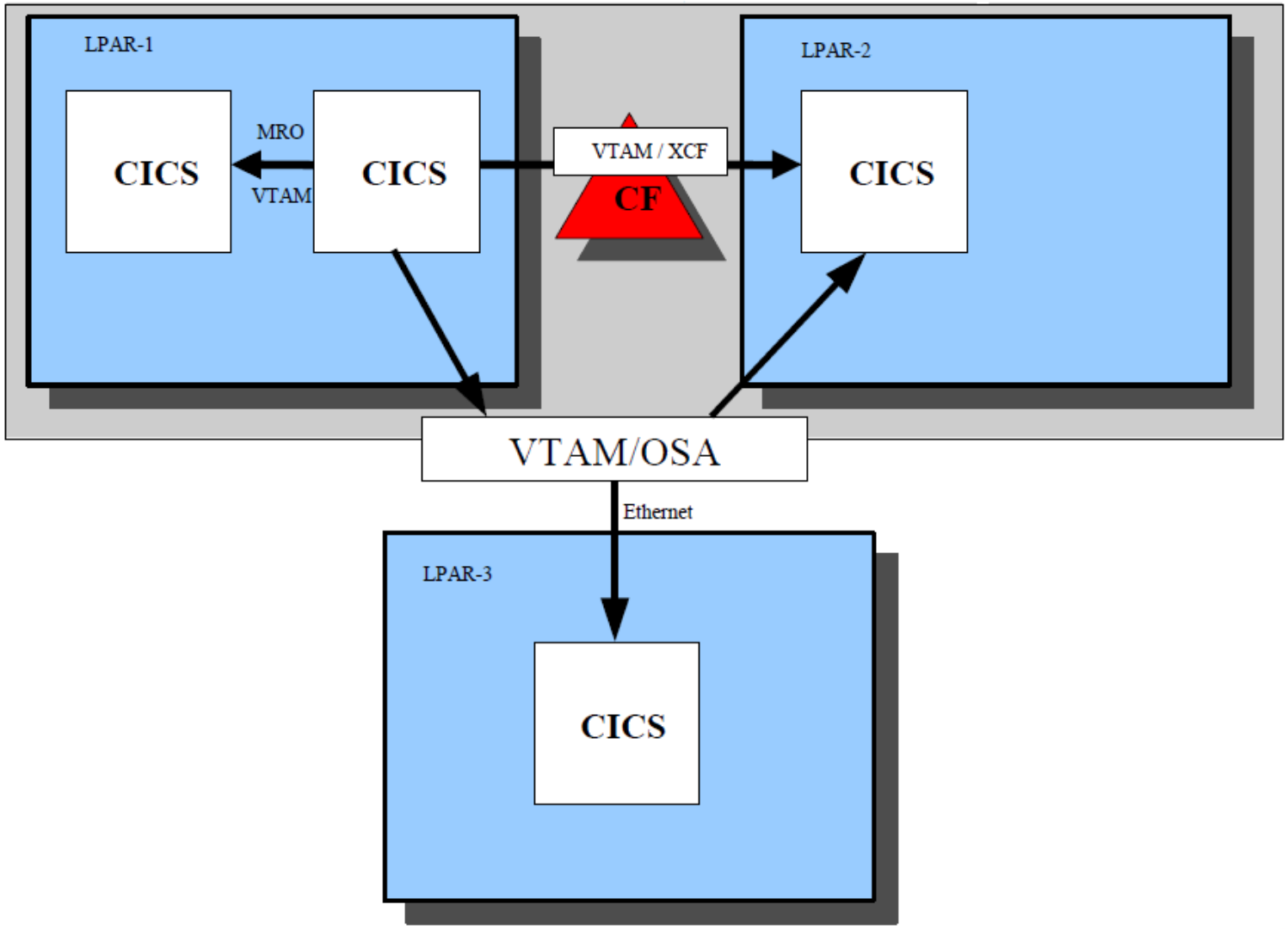
IPIC functions and releases

From	To			
	TXSeries V7.1+	CICS TS V3.2	CICS TS V4.1	CICS TS V4.2
CICS TG V7.1+	DPL (ECI)	DPL (ECI)	DPL (ECI)	DPL (ECI)
TXSeries V7.1+	DPL	DPL	DPL	DPL
CICS TS V3.2	DPL	DPL	DPL	DPL
CICS TS V4.1	DPL	DPL	DPL TR 3270 Async Processing Distributed Identity	DPL TR 3270 Async Processing Distributed Identity
CICS TS V4.2	DPL	DPL	DPL TR 3270 Async Processing Distributed Identity	DPL TR 3270 Async Processing Distributed Identity FS: FC, TD, TS

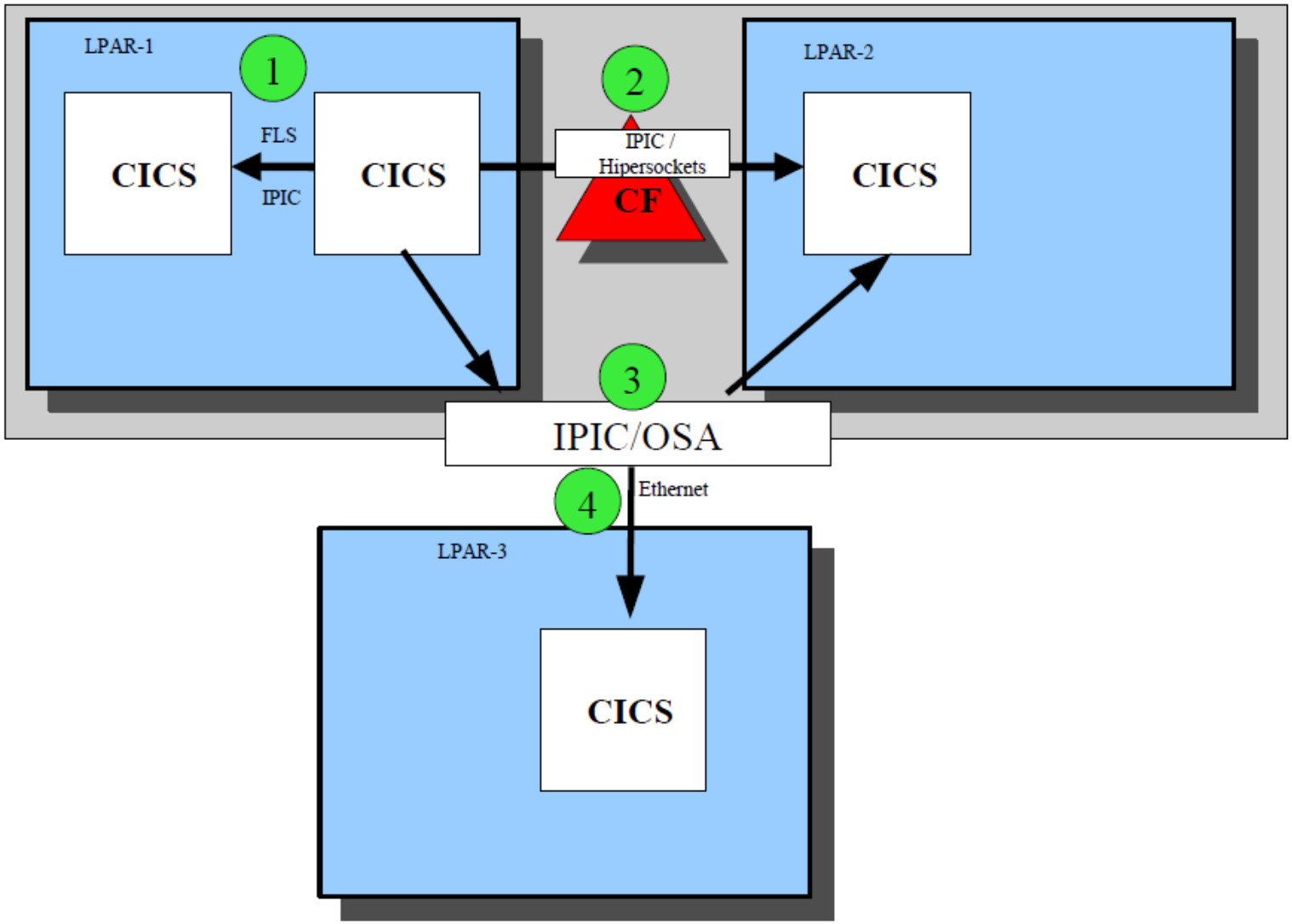
… notes

- Transaction routing of 3270 terminals
  - restricted to **traditional** routing from a CICS region (TOR) that has a unique APPLID
- Asynchronous processing
  - restricted to function shipping of **non-terminal** START, START CHANNEL, and CANCEL commands
- Distributed identity (identity propagation)
  - Supported by CICS TG V8.0 and CICS TS V4
- Function shipping
  - File control
  - Transient data
  - Temporary storage

MRO/ISC connectivity



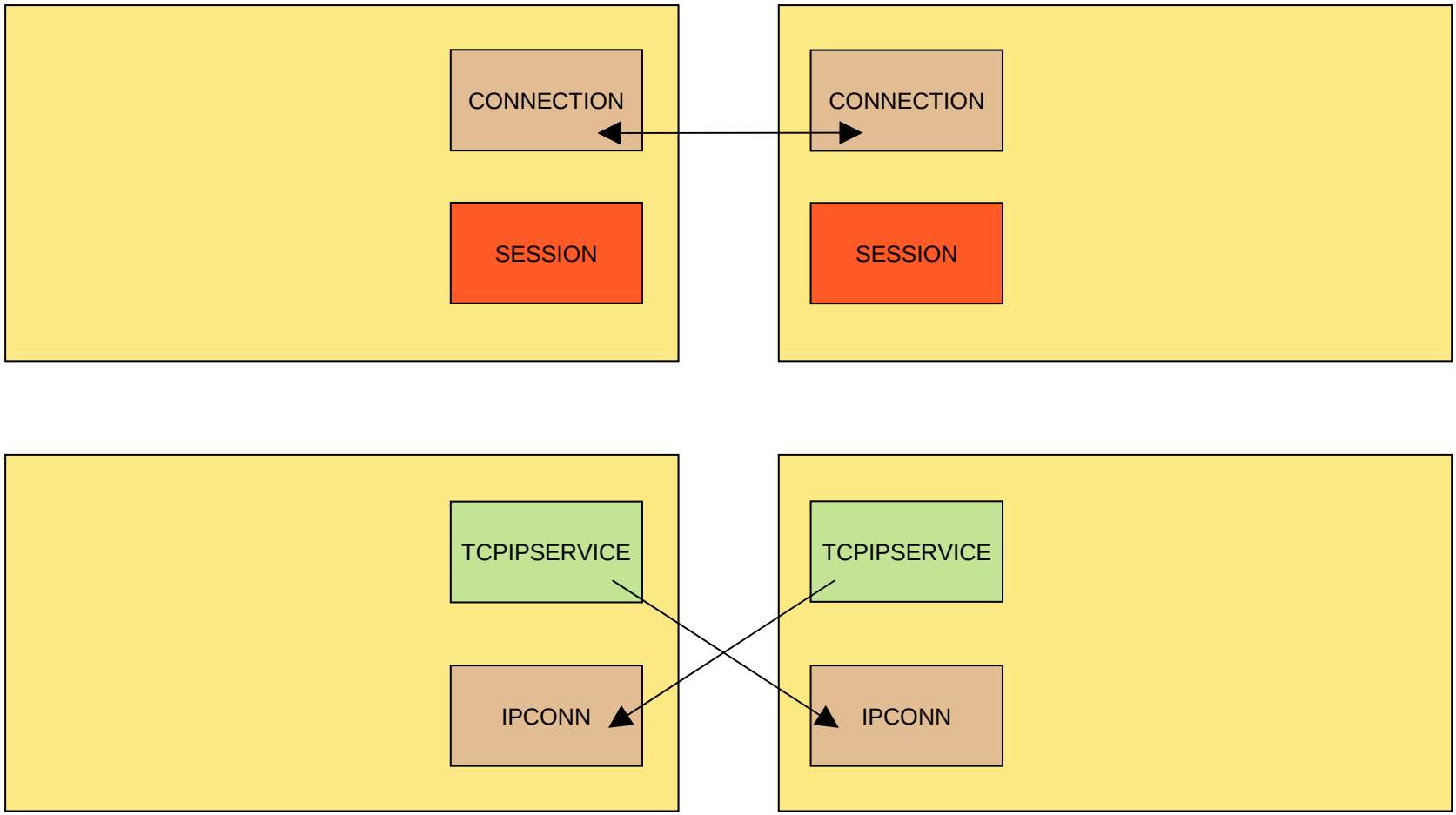
IPIC connectivity



Advantages of TCP/IP

- Open Systems Adapter (OSA) performance
  - OSA-2 and OSA-3 support Queued Direct I/O (QDIO) mode for IP only
- Encryption via Secure Sockets Layer (SSL)
  - Standards based
  - Hardware crypto support
  - Ongoing support of new protocols
- Firewall and other IP security options
- VIPA and WLM support on z/OS
- Socket optimizations in TCP/IP stack
  - Fast Local Sockets
  - Hipersockets

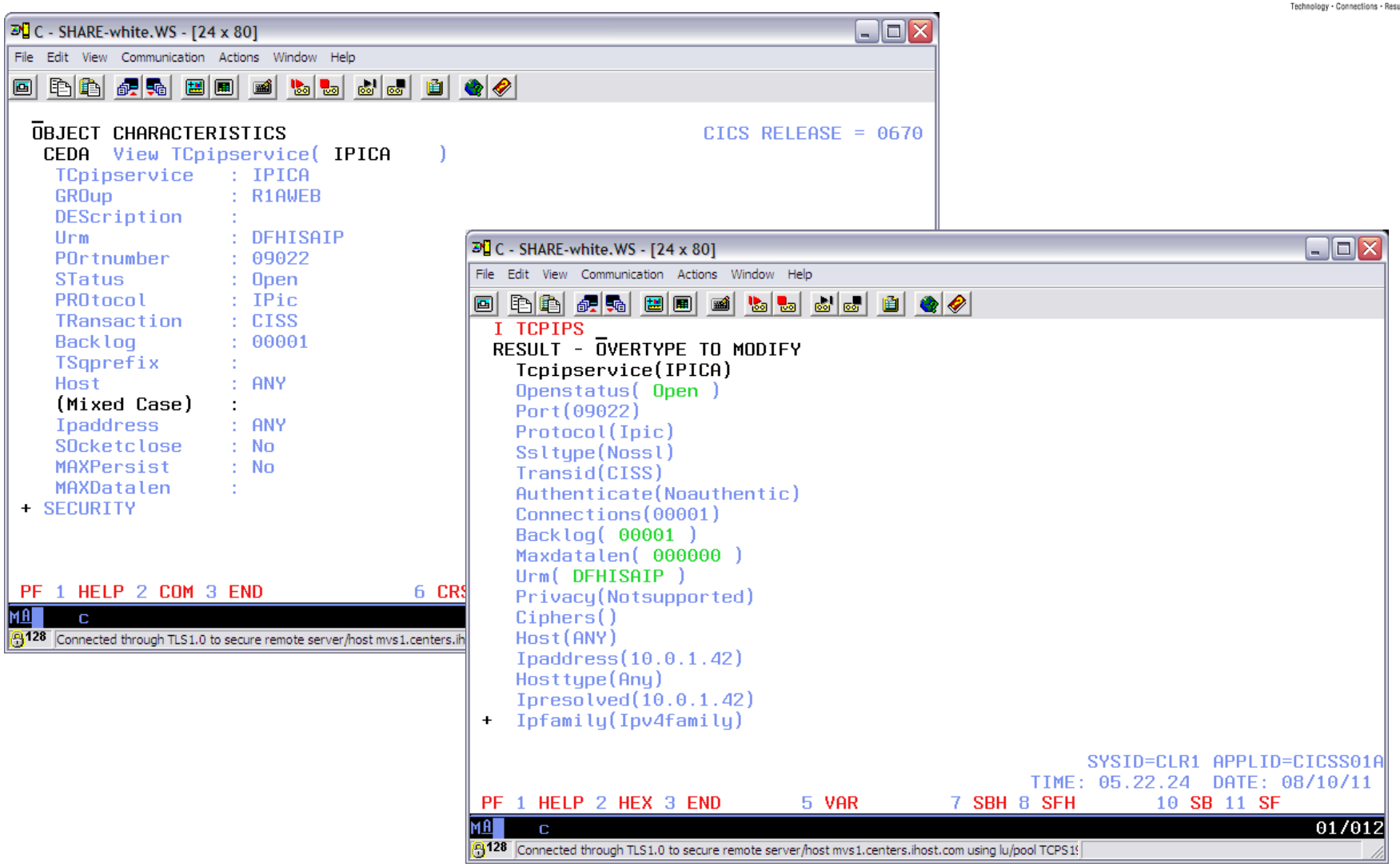
ISC/MRO vs IPIC resources



Resource definitions

- Two related resources
- IPCONN
  - Defines a TCP/IP connection to a remote system
  - Provides attributes of the connection
- TCPIPSERVICE
  - Defines protocol-specific TCP/IP services
  - Specifies port on which CICS will accept IPIC connections
  - Provides some attributes of the inbound connection

TCPIPSERVICE

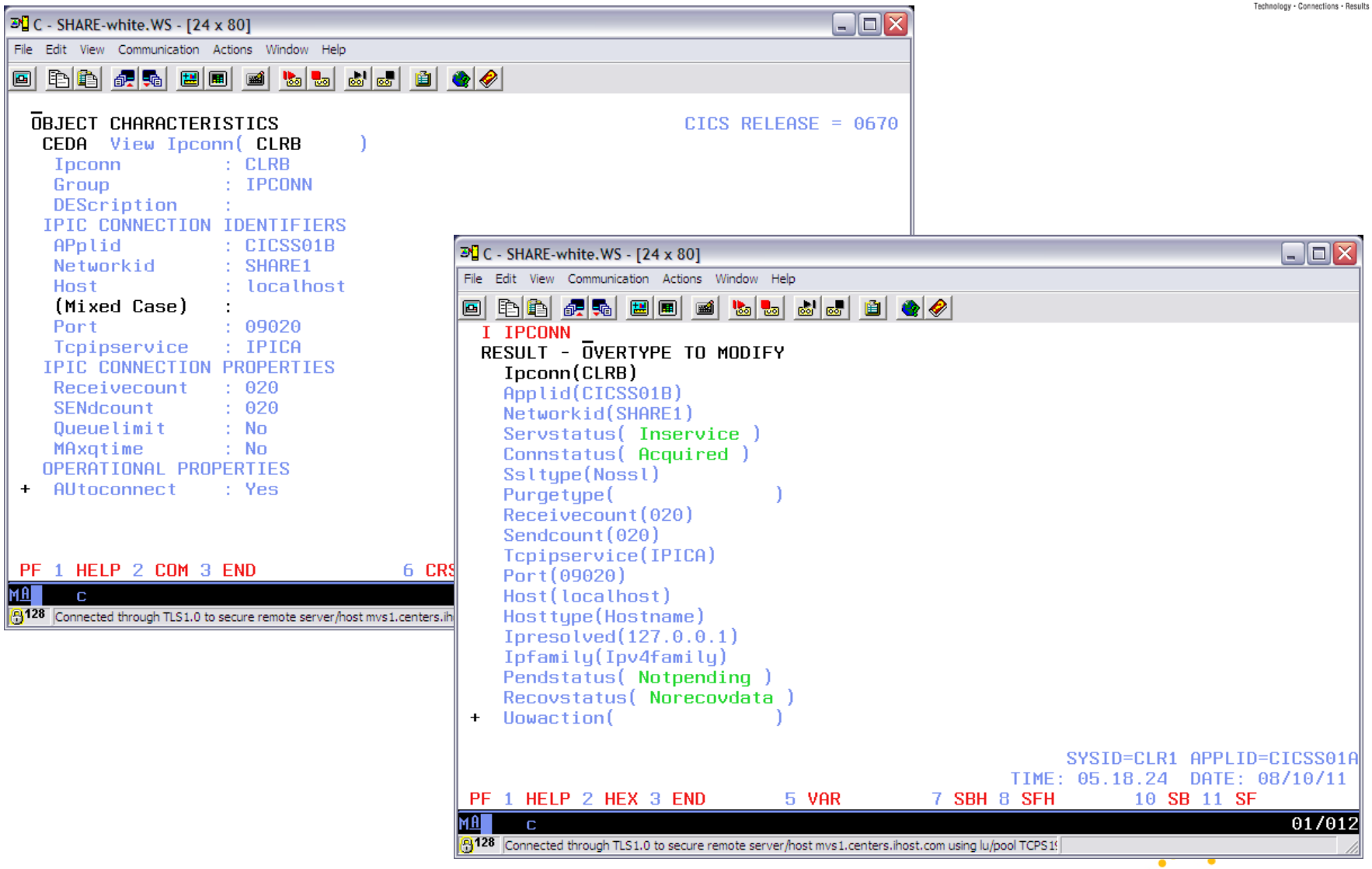


TCPIPSERVICE definition

… notes

- Protocol **=IPIC**
- URM **=DFHISAIP** to autoinstall IPCONN resource, or **=NO** to disable
- HOST / Ipaddress / Portnumber = Listening IP and port
- Transaction **=CISS** which is the CICS transaction to process IPIC requests
- Socketclose **=No**
- SSL **=No**
  - **=Yes** to enables SSL
  - **=Clientauth** to require a client certificate

IPCONN

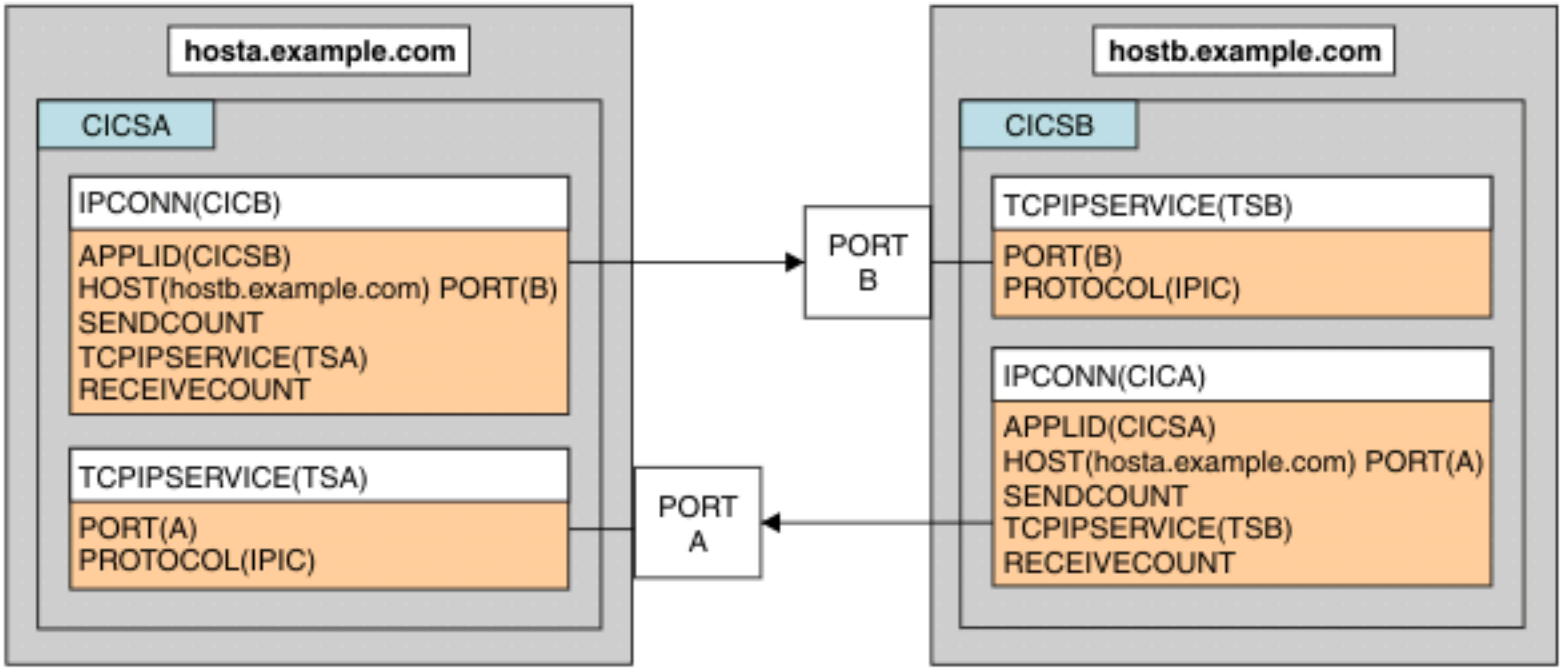

C - SHARE-white.WS - [24 x 80]
File Edit View Communication Actions Window Help
OBJECT CHARACTERISTICS
CICS RELEASE = 0670
CEDA View Ipconn(CLRB)
Ipconn : CLRB
Group : IPCONN
DEScription :
IPIC CONNECTION IDENTIFIERS
APplid : CICSS01B
Networkid : SHARE1
Host : localhost
(Mixed Case) :
Port : 09020
Tcpipservice : IPICA
IPIC CONNECTION PROPERTIES
Receivecount : 020
SENdcount : 020
Queuelimit : No
MAxqtime : No
OPERATIONAL PROPERTIES
+ AUtoconnect : Yes
PF 1 HELP 2 COM 3 END
MA c
Connected through TLS1.0 to secure remote server/host mvs1.centers.ih
C - SHARE-white.WS - [24 x 80]
File Edit View Communication Actions Window Help
I IPCONN
RESULT - OVERTYPE TO MODIFY
Ipconn(CLRB)
Applid(CICSS01B)
Networkid(SHARE1)
Servstatus(Inservice)
Connstatus(Acquired)
Ssltype(Nossl)
Purgetype( )
Receivecount(020)
Sendcount(020)
Tcpipservice(IPICA)
Port(09020)
Host(localhost)
Hosttype(Hostname)
Ipresolved(127.0.0.1)
Ipfamily(Ipv4family)
Pendstatus(Notpending)
Recovstatus(Norecovdata)
+ Uowaction( )
SYSID=CLR1 APPLID=CICSS01A
TIME: 05.18.24 DATE: 08/10/11
PF 1 HELP 2 HEX 3 END 5 VAR 7 SBH 8 SFH 10 SB 11 SF
MA c
01/012
Connected through TLS1.0 to secure remote server/host mvs1.centers.ihost.com using lu/pool TCPS1

IPCONN definition

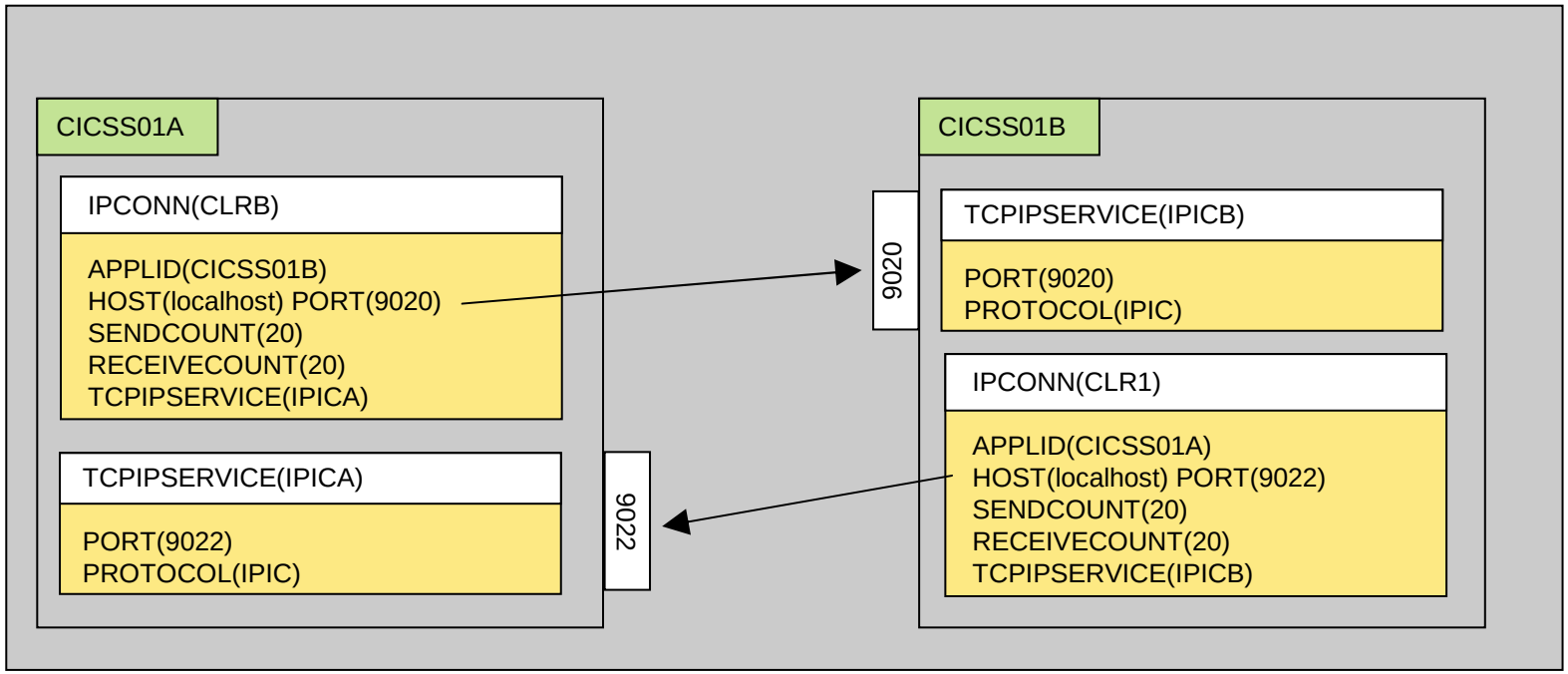
… notes

- Applid = application id of remote CICS region
- Host / Port = listening IP name and port of remote CICS region
- Tcpipservice = TCPIPSERVICE resource name for listening requests
- Sendcount / Receivecount = max number sessions

Example configuration



CICS-to-CICS connections



IPCONN

- Automatically created if using autoinstall
- If between CICS TS regions and between TXSeries and CICS TS
  - CICS establishes two sockets for each way
    - Note CICS TS V3.2 establishes only one socket
  - Up to 4 character IPCONN name
- If between CICS TG to CICS
  - Sendcount **=0** – ie. CICS does not need to open outbound sockets
  - Uses one socket inbound
  - Up to 8 character IPCONN name
- Duplicate IPCONN names with SNA/MRO connections allowed
  - Provides for migration without changing CICS application SYSID

IPCONN autoinstall URM

- TCPIPSERVICE resource defined with
  - PROTOCOL=**IPIC** and URM=**DFHISAIP** | program_name | NO
    - **Autoinstall is enabled by default**
- DFHISAIP sample supplied in SDFHSAMP
  - Optionally select template and attribute values
- Autoinstalled IPCONN resource attributes based on:
  - Information in the IPIC connection request
  - IPCONN template
  - Attribute values returned by the URM
  - CICS-supplied values
- Invoked
  - When TCPIPSERVICE installed
    - Specifies name of installed IPCONN to be used as a template
    - May override APPLID, HOST and PORT
  - When TCPIPSERVICE deleted

User exits

- XISQUE
  - Used to control the number of queued requests for sessions on IPIC connections
  - IPIC equivalent of XZIQUE
- XRSINDI
  - Invoked for install and discard of IPCONN resource
- XISQLCL
  - Used to influence the decision on whether to queue a START NOCHECK request locally against an IPCONN
  - IPIC equivalent of XISLCLQ

Security

- CICS controls
  - Link security and user security
  - Identity context propogation
- TCP/IP controls
  - SSL
  - Firewalls
- RACF controls
  - STACKACCESS and PORTACCESS
    - Limits local users access to specific TCP/IP stacks and ports
  - NETACCESS
    - Limits access to TCP/IP zones

Link and User security

- Link security – how to establish the user ID representing the link / connection
  - Specified in IPCONN LINKAUTH parameter
    - SECUSER
      - *Link user is specified via SECURITYNAME parameter*
    - CERTUSER
      - *Link user is mapped by RACF from the client's SSL certificate*
- Flowed user secuirty - how establish the user ID for the task (attach-time)
  - Specified in IPCONN USERAUTH parameter
    - Local – Tasks run under link user ID or default user ID
    - Identify – User ID is named, but not verified via PW – "Asserted Identity"
    - Verify – User ID and password required and checked
    - Defaultuser - CICS will not accept a user ID and password from the partner system. All requests run under the default user ID

… notes

- 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 in the same sysplex
- If the same sysplex but not using SSL client authentication recommendation is to use TCP/IP NETACCESS controls
- Affects CICS to CICS and CICS TG to CICS requests

SSL support

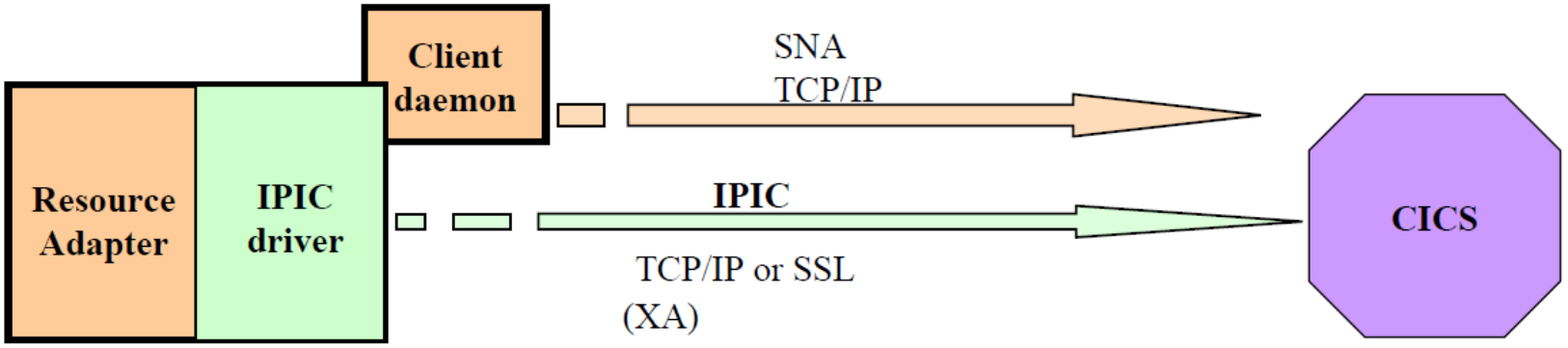
- Digital certificate-based authentication and encryption
- Asserts trust
  - “client” trusts “server”.
  - “server” trusts “client” - also call client authentication - optional
- CICS allows cipher suite selection
  - in IPCONN resource as “client”
  - in TCPIPSERVICE resource as “server”
- Encryption of the connection using strongest common cipher
- Map client certificate to link user ID
  - Specify IPCONN attribute LINKAUTH=CERTUSER

CICS TG-to-CICS connections

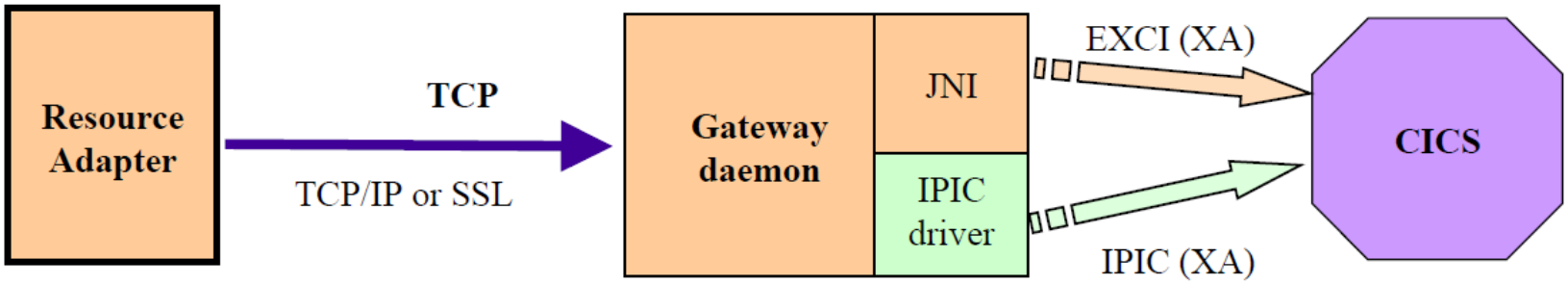
- Simplified Topologies and setup
- Channels + Containers
  - JCA and base classes
  - Data conversion
- XA transaction (two phase commit) support
- SSL connections direct to CICS TS
- New monitoring options: Origin Data
- CICS TG provided IPIC driver pure Java
  - Fully zAAP enabled code path
- Asserted identity options

CICS TG topologies

Multiplatform 2-tier



z/OS 3-tier



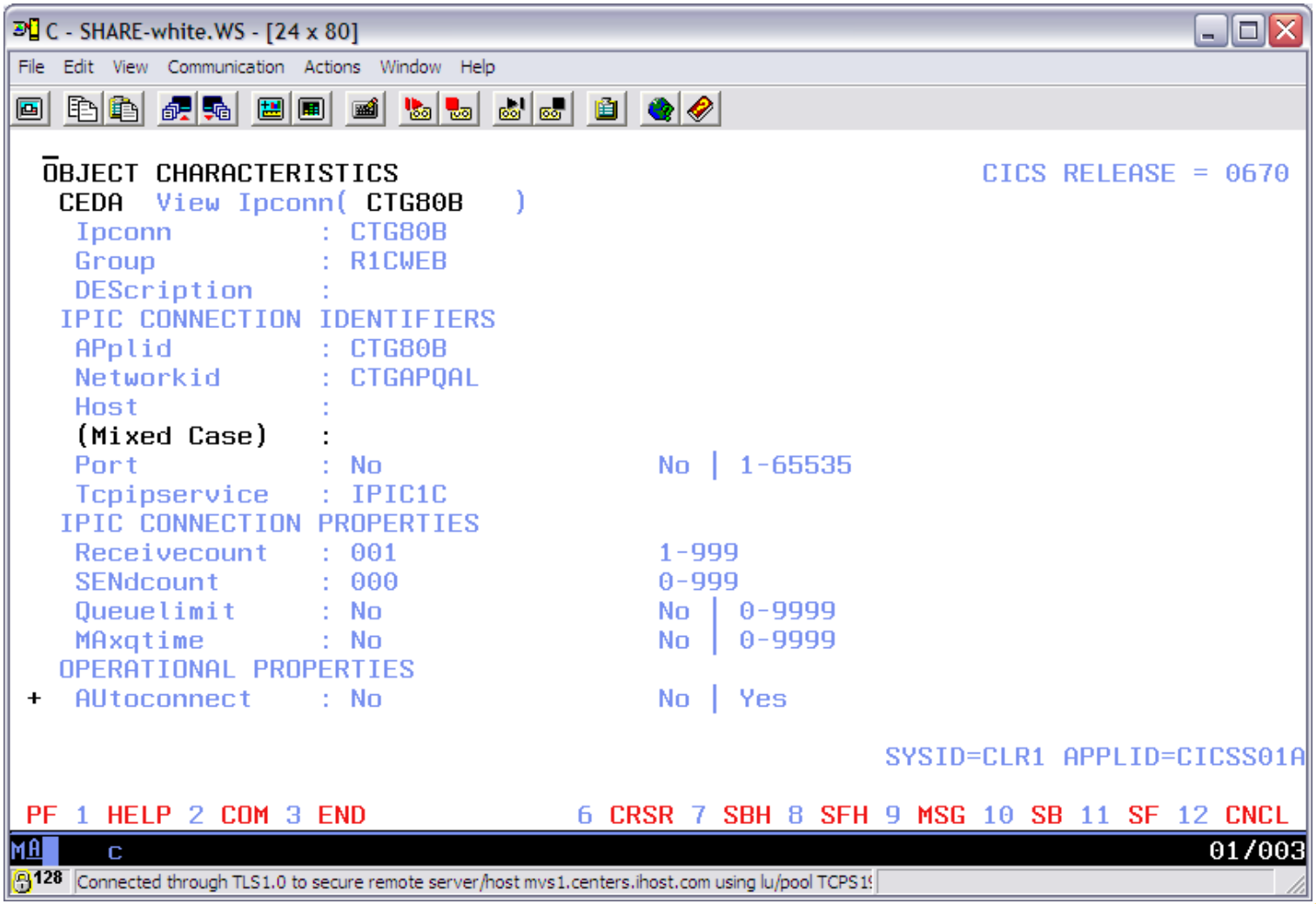
IPIC connections in CICS TG daemon

- Parameters in CICS TG configuration file
  - ctg.ini

```
SECTION PRODUCT
   Applid = CTG80B             # The APPLID of the Gateway daemon
   ApplidQualifier = CTGAPQAL  # The APPLID qualifier of the Gateway daemon
   DefaultServer = CICSS01C    # The default server used by the Gateway daemon
ENDSECTION

SECTION IPICSERVER = CICSS01C   # Arbitrary name for the server
   Description = IPIC Server    # Arbitrary description for the server
   Hostname = 10.0.1.42         # The server's TCP/IP name or IPv4 or IPv6
                                # address
   Port = 9483                  # The listening TCPIPSERVICE port in CICS
   TcpKeepAlive = Y             # Set the TCP KEEPALIVE Socket option
   SrvIdleTimeout = 60          # Time in minutes to keep idle connections open
   ConnectTimeout = 60          # Time in seconds to wait for CICS to respond
                                # during connection establishment
   SendSessions = 100           # Number of simultaneous conversations
                                # requested
   CICSApplid = CICSS01C        # The APPLID of the destination CICS server
                                # (for verification only)
   CICSApplidQualifier = SHARE1 # The APPLID Qualifier of the destination CICS
                                # server (for verification only)
ENDSECTION
```

IPCONN for CICS TG



… notes

- **Applid**: Applid of CICS TG (optional)
  - **NetworkID**: APPLIDQUALIFIER of CICS TG. Required if Applid set
- **Host**: TCP/IP hostname for remote CICS TG(optional) use localhost if on same TCP/IP stack
- **Port: NONE** . Do not specify port for CICS TG one way IPCONN
- **Tcpipservice**: CICS TCPIPSERVICE resource listening for requests
- **Receivecount**: Number of receive sessions to service parallel requests. Set to <= MAXTASKS
- **Sendcount**: **=0** for one-way IPCONNs
- **Queuelimit**: Number of requests to queue waiting for a receive session
- **Autoconnect**: No

Migration planning

- Can migrate existing MRO, APPC, and LUTYPE6.1 connections to IPIC connections
- Existing connections continue to operate as before
- IPCONN takes precedence over CONNECTION definition
  - If an IPCONN and a CONNECTION have the same name, CICS uses the IPCONN
    - If the request is supported over IPIC

Migration Utility

- DFH0IPCC sample program for use with DFHCSDUP system definition utility
- Supply list of APPLIDs with corresponding hostnames and port numbers
- Creates IPCONN and TCPIPSERVICE resources as a series of DEFINE statements, which form the SYSIN for DFHCSDUP

Summary

- IP Advantages
  - Socket optimizations
  - Cost of IP networking
  - Security options
  - IPIC provides increased QoS over SNA
- Multi-release delivery
  - Allows staged migration away from SNA networks
  - Without impacting business applications
- For more details on IPIC, see CICS Information Center
  - IPIC learning path