

Binary for Linux - TCAP

Release Notes for Version 5.08

1. Overview

This release extends the TCAP module to support for up to four Network Contexts. Each Network Context allows TCAP to support a different set of options when used in conjunction with multiple SCCP modules. When used with the SIU mode of the Dialogic® SS7G2x Signaling Gateway and DTS functionality this module can be used to support multiple local point codes and mixed ANSI and ITU-T configurations. This will require V3.00 or later of the SS7G2x binary distribution.

This release is fully backwards compatible with the previous release. Customers not using this module in conjunction with the SIU and DTS functionality need not upgrade from TCAP for Linux V5.07.

2. New Functionality

2.1 Network Context Configuration

A new configuration message TCP_MSG_NC_CONFIG (0x7787) has been added to allow the configuration of additional Network Contexts. The TCP_MSG_CONFIG (0x7780) message should be used to configure the default Network Context for the first network. For each subsequent Network Context the message TCP_MSG_NC_CONFIG is required. The TCP_MSG_NC_CONFIG message contains parameters to define address format and TCAP specific options and therefore allows different behaviour for the module to be selected depending on the Network Context of the dialog. Section 2.2 covers how this is selected using the TCPN_NC parameter.

The meaning of the parameters in the TCP_MSG_NC_CONFIG message is the same as the equivalent parameters in the TCP_MSG_CONFIG message. When used to support multiple local point codes within the same network the options settings should typically be the same in both messages.

TCAP Message TCP_MSG_NC_CONFIG

Synopsis:

Message used to configure the additional Network Context TCAP module for operation.

Message Format:

MESSAGE HEADER		
FIELD NAME		MEANING
type		TCP_MSG_NC_CONFIG (0x7787)
id		Network Context id (value 1 to 3)
src		Sending module_id
dst		TCP_TASK_ID
rsp_req		used to request a confirmation
hclass		0
status		0
err_info		0
len		40
PARAMETER AREA		
OFFSET	SIZE	NAME
0	1	cnf_ver - must be set to zero
1	1	user_id - TC-User module id
2	1	nsap_id - Network-layer module id
3	2	Flags
5	2	max_data
7	1	addr_format
8	32	Reserved for future use, must be set to zero.

Network Context id

The Network Context id will identify the Network Context being defined. The default Network Context (0) is configured using the existing TCP_MSG_CONFIG message therefore this message should only be used for Network Contexts 1 to 3. This assumes that four Network Contexts are permitted.

Note: All other Parameters have the equivalent meaning and types as those used in the TCP_MSG_CONFIG.

Two additional configuration status return values TCPE_INVALID_NC (8) and TCPE_INVALID_VERSION (9) have also been added to indicate inappropriate configuration using the TCP_MSG_NC_CONFIG message. They indicate that the Network

Context is out of range or that the version of the configuration message is inappropriate.

2.2 Network Context Message Handling

When a dialog is initiated remotely no change is required as TCAP will automatically determine which Network Context is appropriate. An indication of the Network Context for the dialog will be passed up to the user in the TCPPN_NC parameter. Where the dialog is initiated locally then the application needs to specify which Network Context the message is destined for. This also indicates which point code will be used as the originating point code.

The Network Context should be indicated in the first message for the dialog being used. In the case of TCAP this will be in the first TCAP Service Request, typically an Invoke Req, using the TCPPN_NC parameter.

If no Network Context is specified then the default Network Context NC0 is assumed.

Parameter name	TCPPN_ NC
Parameter length	Variable, typically 1. Length of zero indicates Network Context is unknown.
Parameter data	Network Context Identifier If the default NC is being used then this parameter is optional. If present it should have a value of 0. For other Network Contexts it should match the value defined in the relevant TCP_MSG_NC_CONFIG message.

Binary for Linux - TCAP

Release Notes for Version 5.09

1. Overview

This release clears a fault in the TCAP Component resource handling that occurs in TCAP host binary V5.07 and V5.08. Customers using these versions are advised to upgrade to this new release.

In other respects this release is fully backwards compatible with the previous release.

2. Faults Cleared

TCAP Component Resource Loss

Under some error cases such as the reception of a Return Error from a Class 4 operation a loss component buffer resource could be experienced. In this situation the user would then see the TCAP Software event TCPSWE_NO_TCPT. This release modifies the component handling to rectify this fault.

Dialogic
02-Feb-06

Binary for Linux - TCAP

Release Notes for Version 6.00

1. Overview

This release expands TCAP to support Long Messages for the support of SCCP Segmentation and Reassembly procedures.

This is the first release since V5.09 and is backwards compatible with that release. This release is developed for use with the SS7 Development Package for Linux V5.00 or later. It cannot be used with earlier development packages.

Customers who wish to make use of the Long Message support offered in V5.00 of the development package should upgrade to this release of software (Long Message support is required for SCCP Segmentation). In addition the customers should upgrade to SCCP for Linux V3.00 or later. Other customers need not upgrade.

2 New functionality

2.1 Support for Long Messages

Starting with this release TCAP now supports the generation and reception of long SCPPN_USER_DATA in SCPPT_N_UNITDATA_REQ and SCPPT_N_UNITDATA_IND primitives to or from SCCP. Similarly, it supports the generation and reception of long TCPN_COMPONENT parameters in the TCP_MSG_CPT_REQ and TCP_MSG_CPT_IND messages to and from the TCAP-User.

To enable correct operation the TCPF_SEGMENTATION (0x0100) flag of TCP_MSG_CONFIG (0x7780) must be set. The max_data field in the TCP_MSG_CONFIG (0x7780) message should also be increased and the GCT environment must be set up to handle long messages. If the max_data field value is not sufficient then the TCAP Maintenance Indication TCPEV_DATA_LEN_ERR will be generated.

See ga237sss.pdf for more details on the new encoding scheme allowing to support parameters longer than 255 bytes.

The TCAP module will fail to initialize if it is configured to support large messages (TCPF_SEGMENTATION flag set), but GCT has not been configured to support large messages.

2.2 Use of Linux shared object

This release makes use of shared object version of the GCT library included in V5.00 of the development package.

Dialogic
10-Mar-06

Binary for Linux - TCAP

Release Notes for Version 6.02

1. Overview

This is the first release since V6.00 and is fully backward compatible with that release. This release corrects the handling of the Application Context parameter when used with multiple Network Contexts. Users with configurations not utilising multiple Network Contexts need not upgrade.

In other respects this release is fully backwards compatible with the previous release.

2 Changes

2.1 Application Context parameter handling for Network Contexts

The previous release would not correctly interpret the Application Context parameter (TCPPN_APPL_CONTEXT) when used with multiple Network Contexts. When handling this parameter the settings for Network Context 0 would always be used. This could lead to messages containing Application Context values which are valid in other Network Contexts being rejected. This release correctly handles this parameter for all Network Contexts.

Dialogic
26-Oct-06

Binary for Linux - TCAP

Release Notes for Version 6.03

1. Overview

This release enhances the handling of TCAP timers to maintain TCAP dialogs and their status when the connection to a host is no longer available. In addition, 8-bit Sequence Control Parameters and Destination Transaction IDs used in the routing of SUA messages are now supported.

This release also corrects the handling of the Application Context parameter when used with multiple Network Contexts. Users with configurations not utilising multiple Network Contexts need not upgrade.

This release is fully backwards compatible with the previous release.

2 New Functionality

2.1 Enhanced timer handling

The timer functionality has been enhanced and now enables pending TCAP dialogs and their status to be maintained when the connection to a host is lost.

For each new dialog, a per-dialog timer will now be initiated only if it has been specified by the user. The timeout value can be set by including the new optional parameter, TCPPN_DLG_IDLE_TIMEOUT (0x16), in the TC-BEGIN message.

Parameter name	TCPPN_DLG_IDLE_TIMEOUT 0x16
Parameter length	Fixed, set to 2.
Parameter data	Length of timeout for Dialogue. Double octet, set to 0 or a number to specify the duration for that Dialogue in timer ticks. 0 = No timer is run

A new TCAP module option, TCPF_DLG_TIM_ABORT (0x0200), has been added to the Module Configuration Request message (0x7780) to optionally determine the behaviour of the module when a timer expires. If this option is set, this will automatically close and subsequently abort the dialog on timer expiry.

If the TCPF_DLG_TIM_ABORT option is not enabled, the module will send a Maintenance Event Indication (0x07a1) with status 0x08, to

notify the user that the affected timer has expired. Additionally, the timer will be reset to the initial timeout value.

Support for 8-bit Sequence Control Parameters

This TCAP release now supports the generation of 8-bit Sequence Control values in messages to SCCP (in the SCPPN_SEQ_CTRL parameter). This is to allow SCCP to make use of eight bits when generating SLS values.

Support for SUA

A new TCAP module option, TCPF_DTID_ON (0x0400), has been added to the Module Configuration Request message (0x7780). If the TCPF_DTID_ON option is set, this will enable messages to be routed using the Destination Transaction ID (DTID) which is required for SUA based configurations.

3 Changes

3.1 Application Context parameter handling for Network Contexts

The previous release would not correctly interpret the Application Context parameter (TCPPN_APPL_CONTEXT) when used with multiple Network Contexts. When handling this parameter the settings for Network Context 0 would always be used. This could lead to messages containing Application Context values which are valid in other Network Contexts to be rejected. This release correctly handles this parameter for all Network Contexts.

Dialogic
21-Dec-07

Binary for Linux - TCAP

Release Notes for Version 6.04

1. Overview

This release corrects and changes the Dialogue Idle Abort operation and fixes a number of other faults. Users with configurations not utilising the Dialogue Idle Abort feature or not requiring 8-bit SLS values need not upgrade.

In other respects this release is fully backwards compatible with the previous release.

2. Changes

2.1 Correction to Dialogue Idle Abort operation

The previous release would only abort a maximum of 2 dialogues per 100ms tick when checking for Idle Timeout. Further dialogues (which should have been aborted on the tick) were not aborted.

This release will abort a maximum of 10 dialogues per tick for Idle Timeout. When more than 10 dialogues timeout on the same tick, further dialogues over 10 will have their timeouts extended and spread out over the next 2 seconds.

This change corrects the fault and prevents systems being overwhelmed by dialogues being aborted on the same tick.

When the module starts to defer Idle Timeouts (as above), it will send a new event TCPEV_EXCESSIVE_DLG_ABORTS,(0x0009) to the Maintenance module in a TCP_MSG_MAINT_IND (0x07a1) message. This event will only be sent once per tick. (Note that the maintenance event TCPEV_DLG_TIM_TIMEOUT (0x0008) continues to be sent each time a dialogue is aborted.)

2.2 Changes to Dialogue Idle Timeout operation

The TCPPN_IDLE_DLG_TIMEOUT parameter has been changed to specify the Dialogue Idle Timeout value in seconds (the previous release used 100ms 'ticks'). This brings the parameter in line with the other TCAP timeout parameter TCPPN_TIMEOUT.

The TC-User can now define the Dialogue Idle Timeout parameter TCPPN_DLG_IDLE_TIMEOUT when sending TCPPT_TC_CONTINUE for an established dialogue. This allows the user to change the timeout duration for outgoing or incoming dialogues. The user can also cancel an established idle timeout by setting the parameter to zero or set a new idle timeout where no timeout was initially set.

2.3 SCPPN_SEQ_CTRL values now use 8 bits

This release sends 8-bit SLS values to SCCP using the SCPPN_SEQ_CTRL parameter (if Sequence Control is required). The SLS value used is defined either directly by the TCPPN_QOS parameter from the TC-User (see TCAP Programmer's Manual) or internally by the TCAP module (from the least significant 8 bits of the Dialogue Reference).

Note: From this release, if Sequence Control is enabled, TCAP will always send SCPPN_SEQ_CTRL parameter values in the range 0x00 to 0xff to the SCCP module.

2.4 TC-UABORT uses Dialog Portion ABRT APDU

Previous releases failed to include an ABORT APDU Dialog Portion in the message sent to SCCP when the TC-User sends a TC-UABORT and TCAP module is in the 'Initiation Sent' state. This caused the receiving TCAP to reject the U-ABORT (the user was sent a P-ABORT instead). This release corrects the fault and sends an ABORT APDU Dialog Portion in the SCCP message when in this state.

Dialogic
04-Jul-08

Binary for Linux - TCAP

Release Notes for Version 6.05

1. Overview

This release adds enhancement to the configuration of dialogue ranges now allowing the full available dialogue range to be allocated in any way between incoming and outgoing dialogues ranges.

In addition, this release corrects the configuration and use of Dialogue Groups. Users that do not make use of Dialogue Groups need not upgrade.

In other respects this release is fully backwards compatible with the previous release.

2 New Functionality

2.1 Enhancement of Dialogue Ranges

The configuration of dialogue ranges has been enhanced by removing the restriction that had previously required the most significant bit to be set for incoming user dialogues. TCAP no longer inspects whether the most significant bit is set in the 'base_icdlg_id' parameter (offset 18) in TCAP Configuration Request message (TCP_MSG_CONFIG, 0x7780) and that all values for the configured outgoing range are less than 0x8000. This allows the full available dialogue range (0x0000 to 0xffff) to be allocated in any way between incoming and outgoing dialogues ranges.

If any of the above conditions are not met the TCAP Configuration Request message, (TCP_MSG_CONFIG, 0x7780), will be rejected with status TCPE_BAD_PARAM (0x06).

2 Changes

2.1 Allow zero sized incoming or outgoing Dialogue Group ranges

The previous releases could fail if a Dialogue Group range was set to zero. This release allows either the incoming or the outgoing range to be set to zero. As before, if both ranges are set to zero the Dialogue Group is rejected.

2.2 Active dialogue count in Read TCAP Module Status Request for Dialogue Groups

In previous releases, the Read TCAP Module Status Request message (TCP_MSG_R_MOD_STATUS, 0x6796) did not count of any active

incoming dialogues allocated to Dialogue Groups in the 'num_ic_dlg' field when TCAP was interrogated for resource information.

This release corrects the reporting of the 'num_ic_dlg' field in the confirmation message (0x2796) and now indicates the total count of active incoming dialogues across the TCAP module including dialogues allocated to Dialogue Groups.

Dialogic
06-Nov-08