

Source: T3

Title: Change Requests on USAT Interpreter (TS 31.112 / TS 31.113)

Document for: Approval

This document contains change requests as agreed by T3.

T3 Doc	Spec	CR	Rel	Cat	Subject
T3-010738	31.112	001	Rel-5	F	Correction of TAR value usage
T3-010739	31.113	001	Rel-5	F	Addition of SendAdditionalInformation attribute
T3-010740	31.113	002	Rel-5	F	Collection of clarifications
T3-010741	31.113	003	Rel-5	C	Changes to USAT Interpreter system information partition table
T3-010755	31.113	004	Rel-5	B	Comparison with a variable value

CR-Form-v3

CHANGE REQUEST

⌘ **31.112 CR 001** ⌘ rev **-** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Correction of TAR value usage		
Source:	⌘ T3		
Work item code:	⌘ USAT Interpreter	Date:	⌘ 06-11-2001
Category:	⌘ F	Release:	⌘ REL-5
	Use <u>one</u> of the following categories: F (essential correction) A (corresponds to a correction in an earlier release) B (Addition of feature), C (Functional modification of feature) D (Editorial modification)		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)
	Detailed explanations of the above categories can be found in 3GPP TR 21.900.		

Reason for change:	⌘ The distinction between different types of access is made by TAR value ranges rather than by dedicated TAR values.
Summary of change:	⌘ Usage of the term "TAR value range" instead of "TAR value".
Consequences if not approved:	⌘ Inconsistency between the series of Interpreter specifications

Clauses affected:	⌘ 4.3, 6.4	
Other specs affected:	<input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘
Other comments:	⌘	

4.3 USAT Interpreter System Architecture

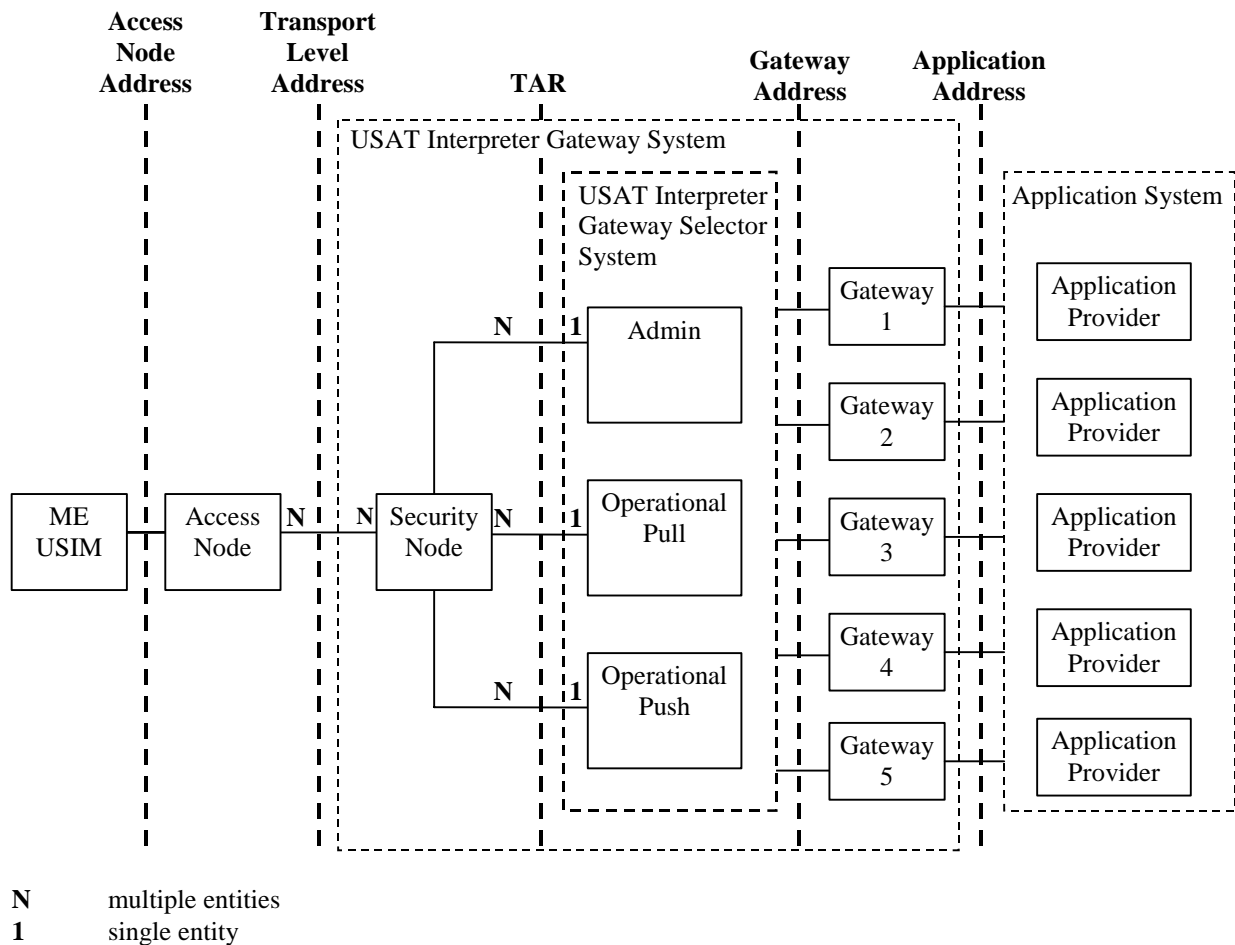


Figure 1: System Architecture

The Access Node is the network entity that provide the transport layer specific connectivity to the Security Node as specified in TS 23.048 [4]. The Access Node can be connected to any number of different Security Nodes.

The bearer type determines the access node. For example in the case where SMS is the bearer, the Access Node would be an SMSC. In GPRS the Access node would be a GGSN. The Access Node is addressed according to the bearer's addressing principle. The user reaches the Access Node using the Service Center Address if the bearer is SMS. If the bearer is IP, the user addresses the Access Node using its IP Address.

The Security Node is the entity that terminates the 23.048 protocol. The Security Node is addressed according to the used bearer. For example in the case where SMS is used as bearer, the Access Node addresses the Security Node using the Destination Address. In the case where the bearer is IP, the IP Port addressing is used to reach the Security Node.

The Gateway Selector is the entity that subscribes to data from the Security Node based on TAR value and is responsible for connecting the data flow into the appropriate Gateway for the application that is addressed.

The Gateway Selector System consists of logically separate Gateway Selectors to handle the different types of access. These are Administrative, Operational Pull and Operational Push Access. The distinction between these is made using separate TAR value [range](#)s. Thus, one TAR value [range](#) is reserved for each of these three different access types. The TAR value [range](#)s are specified in TS 31.114 [3].

The Gateway is the entity that has the capability to encode and decode data between the formats used by the application system and the USAT Interpreter byte codes. The Gateway terminates the operational layer of the protocols. One Gateway potentially handles only a limited set of conversions from Application encoding to USAT Interpreter byte codes. There might be Gateways for dedicated purposes that can be addressed using the Gateway Address. Examples can be separate Gateways for banking, different application languages, content types etc.

6.4 Administrative mode

The following figure gives an example for a data exchange in the administrative mode.

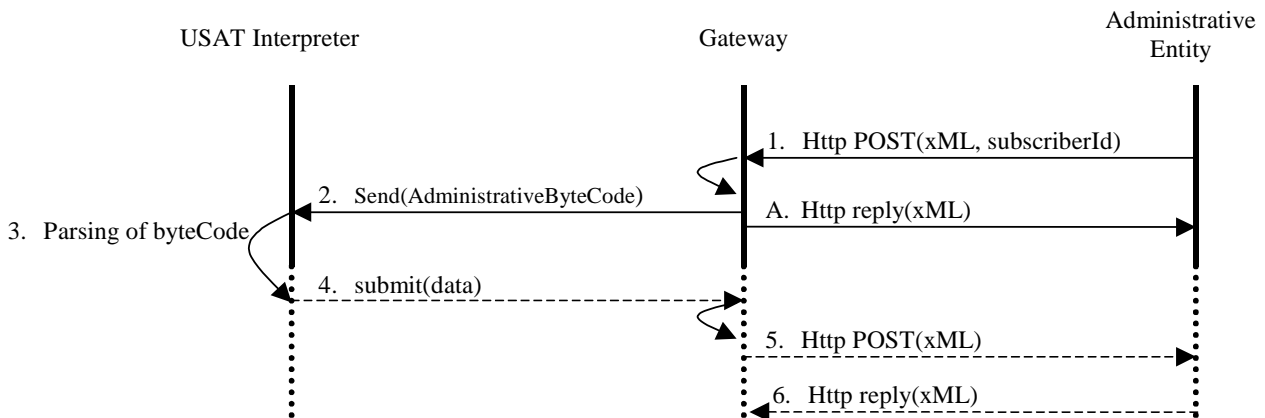


Figure 2: USAT Interpreter Administrative Flow

1. The Administrative Entity requests the Gateway to send administrative byte codes to the USAT Interpreter. In the given example, this request uses Http.

parallel and independent processing:

<p>2. The Gateway sends administrative byte codes to the USAT Interpreter using the transmission protocol for administrative messages (TS 31.114 [3]).</p>	<p>A. The Gateway optionally replies data after the interpretation of the information received from the Administrative Entity. In the given example, where the Http protocol is used, the data reply of the Gateway is mandatory.</p>
<p>3. The USAT Interpreter renders the received administrative byte codes. In the given example, where no blocking conditions are defined, the delivered administrative byte code from the Gateway is rendered.</p>	
<p>4. If the USAT Interpreter encounters a reply request within the administrative byte codes, the USAT Interpreter shall send information using the transmission protocol (TS 31.114 [3]) to the Gateway.</p>	
<p>5. The Gateway shall process the information previously received from the USAT Interpreter and then forward the resulting information to the Administrative Entity.</p>	
<p>6. In the given example, where the Http protocol is used, the Http reply to the Administrative Entity is mandatory.</p>	

The logic of the administrative flow is similar to the Push Mode from the previous clause. The difference is that the USAT Interpreter is addressed through a TAR [value range](#) that has been reserved for administrative commands. The behaviour of the administrative mode depends on the state of the USIM Interpreter at reception of the USAT Interpreter byte code.

CR-Form-v3

CHANGE REQUEST

⌘ **31.113 CR 001** ⌘ rev **-** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Addition of SendAdditionalInformation attribute to the Gateway Address TLV		
Source:	⌘ T3		
Work item code:	⌘ USAT Interpreter	Date:	⌘ 06-11-2001
Category:	⌘ B	Release:	⌘ REL-5
	Use <u>one</u> of the following categories: F (essential correction) A (corresponds to a correction in an earlier release) B (Addition of feature), C (Functional modification of feature) D (Editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

Reason for change:	⌘ A new requirement for the Gateway Address TLV specified in TS 31.113 has been identified.
Summary of change:	⌘ A new optional attribute "SendAdditionalInformation" is specified for the Gateway Address TLV. This allows the application to select the appropriate Gateway.
Consequences if not approved:	⌘ To support mechanisms which are specified in TS 31.114 (Draft).

Clauses affected:	⌘ 7.9.3.4, 13	
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘
Other comments:	⌘	

7.9.3.4 Gateway Address

The Gateway Address TLV contains data (the Gateway Address Information) to address a specific Gateway in the USAT Interpreter Gateway System. The coding of the Gateway Address Information is out of the scope of the present document.

If the Gateway Address TLV is available in the Submit Configuration TLV, the USAT Interpreter shall put the Gateway Address Information into the Gateway Address TLV of the operational layer of the USAT Interpreter transport protocol (see TS 31.114 [2]).

Length	Value	Description	M/O
1	'15' / '95'	Gateway Address Tag	M
1-3	A+B	Length	M
A	Data	Attributes	O
BA	Data	Gateway Address Information	O

Attributes:



13 Tag Values!end!

The present document uses the following Tag values:

Tag Value	Usage
'01' / '81'	Page Tag
'02'	Page Identification Tag
'03'	Page Unlock Code Tag
'04'	One Time Password Tag
'05'	Keep Alive List Tag
'06'	Service ID Tag
'07'	String Pool Tag
'08' / '88'	Terminal response handler modifier tag
'09' / '89'	Action TLV tag
'0A' / '8A'	Navigation Unit Tag
'0B'	Anchor Tag
'0C'	Anchor Reference Tag
'0D'	Variable Identifier List Tag
'0E' / '8E'	Inline Value Tag
'0F' / '8F'	Inline Value 2 Tag
'10'	Input List Tag
'11'	Ordered TLV List Tag
'12' / '92'	Page Reference Tag
'13' / '93'	Submit Configuration Tag
'14'	Submit Data Tag
'15' / '95'	Gateway Address Tag
'16'	Submit Tag
'17' to '3F'	RFU for data structures
'40'	Set Variable Tag
'41'	Assign and Branch Tag
'42'	Extract Tag
'43' / 'C3'	Go Back Tag
'44'	Branch on Variable Value Tag
'45' / 'C5'	Exit Tag
'46' / 'C6'	Execute USAT Command Tag
'47' / 'C7'	Execute Native Command Tag
'48'	Get Length Tag
'49'	Get TLV Value Tag
'4A' / 'CA'	Display Text Tag
'4B' / 'CB'	Get Input Tag
'4C' to '7F'	RFU for commands

All other Tag values are RFU.

CR-Form-v3

CHANGE REQUEST

⌘ **31.113 CR 002** ⌘ rev **-** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Clarifications to TS 31.113		
Source:	⌘ T3		
Work item code:	⌘ USAT Interpreter	Date:	⌘ 06-11-2001
Category:	⌘ F	Release:	⌘ REL-5
	Use <u>one</u> of the following categories: F (essential correction) A (corresponds to a correction in an earlier release) B (Addition of feature), C (Functional modification of feature) D (Editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

Reason for change:	⌘ The clarifications are provided to enhance the readability of the specification and to minimise the risk of ambiguities.
Summary of change:	⌘ Several changes to the text.
Consequences if not approved:	⌘ A risk of ambiguities in the specification.

Clauses affected:	⌘ 4.2.3, 4.3, 5.3, 7.1.8.2, 7.1.8.3, 7.1.8.4.1, 7.1.8.4.3, 7.1.8.4.4, 7.2.2, 7.3, 7.9.3.1, 7.10, 7.10.1, 7.10.2, 8.2.3, 8.7.4, 8.7.5.2, 8.8.2, 9	
Other specs affected:	<input type="checkbox"/> Other core specifications ⌘ <input type="checkbox"/> Test specifications ⌘ <input type="checkbox"/> O&M Specifications ⌘	
Other comments:	⌘	

4.2.3 Wait State

When rendering a Page Reference TLV containing a Submit Configuration TLV having the "ProcessingBehaviour" attribute set, the USAT Interpreter shall perform the following actions:

- provide the Submit TLV to the protocol layer to be transmitted to the external system entity (see clause 4.2.2)
- process next byte code.

When rendering a Page Reference TLV containing a Submit Configuration TLV having the "ProcessingBehaviour" attribute not set, the USAT Interpreter shall perform the following actions:

- Generate a new RequestID value, by incrementing the RequestID value. If the Request ID value reaches its maximum value, the RequestID value shall start at 0 again.
- Provide the RequestID to the protocol layer to be incorporated into the transport protocol (refer to 3GPP TS 31.114 [2]).
- Provide the Submit TLV to the protocol layer to be transmitted to the external system entity (see clause 4.2.2).
- enter the wait state.

In the wait state, the USAT Interpreter shall keep the proactive session alive. Therefore, a DISPLAY TEXT USAT command shall be issued by the USAT Interpreter to notify the user that the USAT Interpreter has entered the wait state.

The text to be used for the text string of the DISPLAY TEXT command shall be taken from the Inline Value TLV of the Submit Configuration TLV requesting the wait state.

If this Inline Value TLV is not available in the Submit Configuration TLV when entering the wait state, then a default text shall be taken by the USAT Interpreter. This default text can be personalised and later on changed by administrative means.

For the DISPLAY TEXT USAT command the command qualifier option:

- "clear message after delay".

shall be used.

...

4.3 Terminal response handler mechanism

For any general result of an USAT command, the USAT Interpreter shall branch to the terminal response handler. The terminal response handler will match the general result with the currently defined general result ranges and process the corresponding action.

If several actions are assigned to a general result range, a SELECT ITEM command shall be built [by the USAT Interpreter](#) from the action list using the action description ~~by the USAT Interpreter~~ to allow the user to choose between actions.

In case of an exception of the USAT Interpreter or no corresponding general result range to the general result of an USAT command, a default action will apply. This default action could be changed by using the terminal response handler modifier with the reserved general result range 'FF FF'.

Exception example:

- no more byte code when process next byte code (e.g. end of navigation unit).

A default terminal response handler configuration is defined in the present document and can be modified by administrative means or at personalization stage.

The default terminal response handler configuration can be modified temporarily by the terminal response handler modifier (e.g. to hide default entries by using action IDs, to add new ones or to modify existing entries).

If the USAT Interpreter branches to another page due to the terminal response handler configuration, the standard inter page variable management shall apply ([see clause 6.1.3.1](#)).

Default terminal response handler configuration.

...

5.3 Coding of attribute bytes

The MSB of each attribute byte indicates if another attribute byte follows or not. The MSB is called follow bit. The remaining seven bits of an attribute byte contain TLV specific attributes, either coded as a single bit or as a combination of consecutive bits.

The context, namely the tag, completely determines the order, span and semantics of the bit-packed attribute values. An attribute consisting of more than 1 bit may span two attribute bytes ~~but this situation should be avoided if possible.~~

General coding:

7.1.8.2 General result range

A general result range defines subsequent values of the general result in the terminal response to an USAT command.

- A range consisting of only one value of the general result is coded by setting both bytes to the desired value.
- A range is coded by setting the first byte to the lowest value of the range and the second byte to the highest value of the range.

For example:

- general result '10' shall be coded: '10 10';
- general result '1X' shall be coded: '10 1F';
- general result 'XX' shall be coded: '00 FF';
- general result between '11' and '13' shall be coded: '11 13'.

General result ranges on the USAT Interpreter side shall be checked on a stack based model (by order of occurrence in the byte code; ~~order of TLVs within the navigation unit, order of TLVs within the page~~), meaning the last defined modification shall be checked first. The action(s) defined for the first matching general result range found; (which contains the received general result of the USAT command), shall be executed.

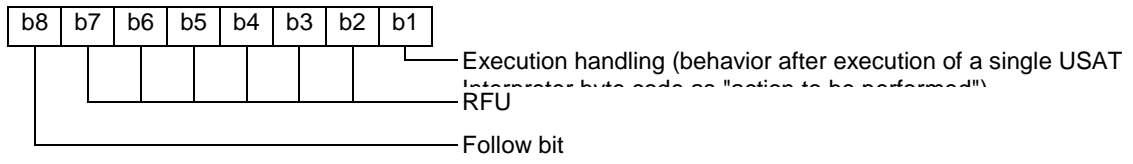
7.1.8.3 Text for user notification

This text is displayed by a DISPLAY TEXT command whenever a general result in response to a proactive command is received, that matches the general result range. After this DISPLAY TEXT command has been issued by the USAT Interpreter the actions defined for the general result range are to be performed regardless of the general result of the DISPLAY TEXT command itself.

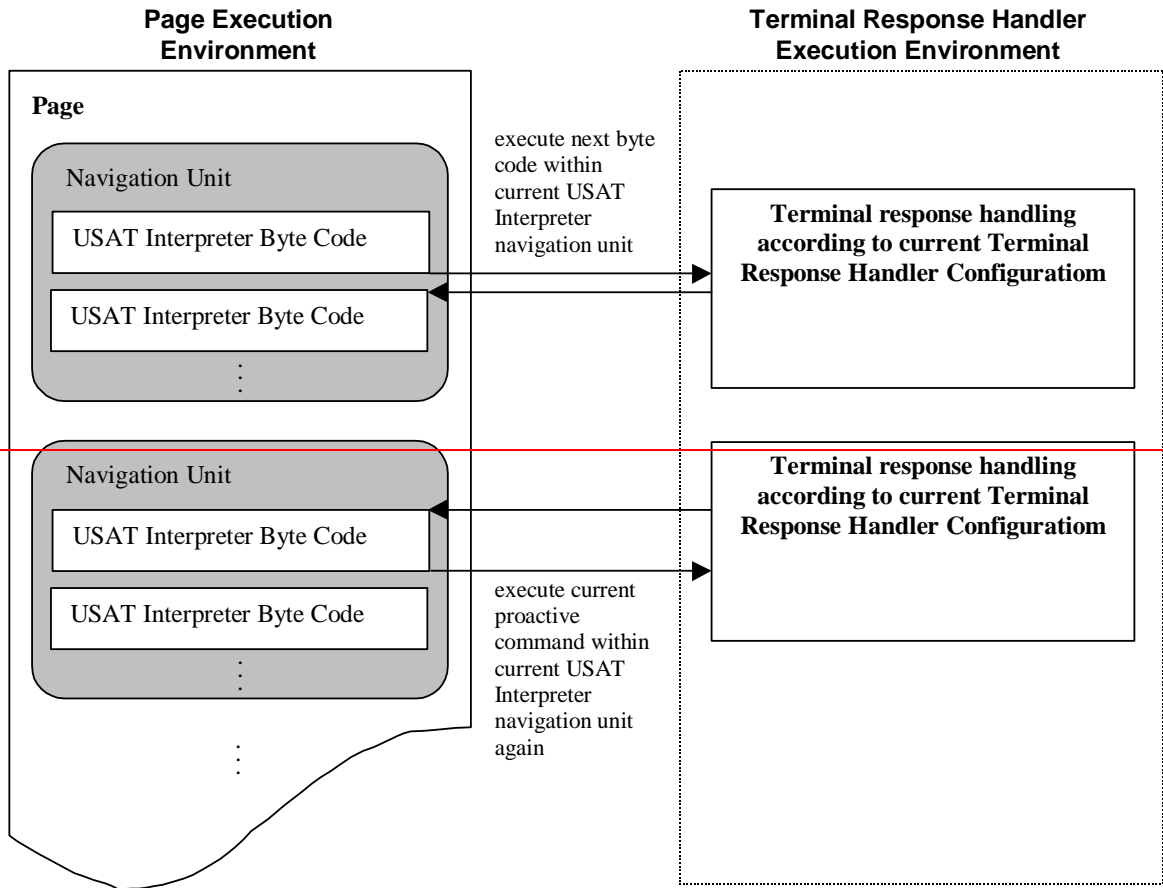
The parameters for the DISPLAY TEXT command shall be as follows:

- The DCS for the DISPLAY TEXT command shall be set according to the value type information of the Inline Value TLV.
- The command qualifier to be used for the DISPLAY TEXT command shall be '81' (["wait for user to clear message" and "high priority"](#)).

7.1.8.4.1 Attributes



The following figure gives an overview of the return behaviour of the terminal response handler [depending on the attribute value](#).



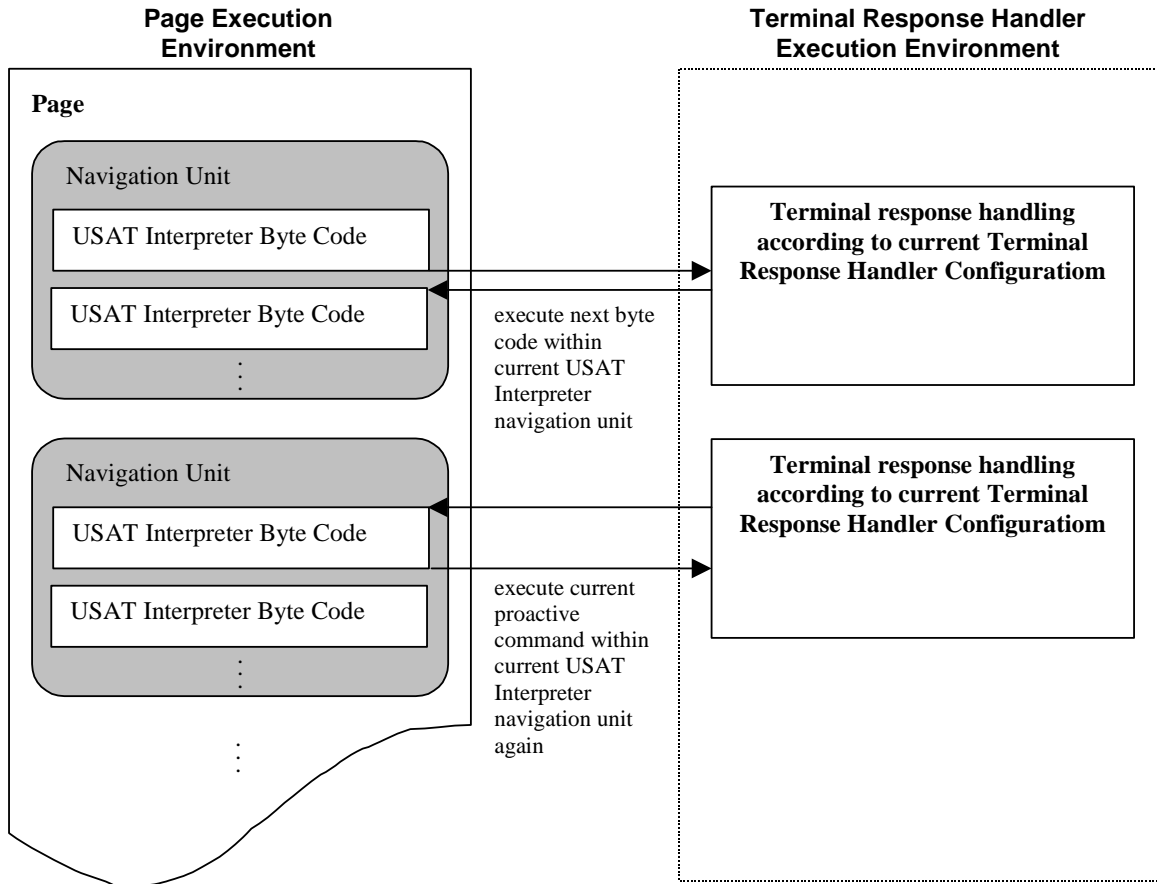


Figure 7.1

[This attribute is to be considered only for certain types of actions to be performed \(see table in clause 7.1.8.4.3\).](#)

7.1.8.4.3 Action to be performed

The action to be performed is either predefined by the USAT Interpreter system (system action) or flow control information (navigation action) or a single USAT Interpreter byte code to be executed.

A system action is indicated within the action TLV by a predefined system action ID only:

- process next byte code;
- quit USAT Interpreter without user confirmation;
- go back one entry in history list. If the history list is empty, this action shall be ignored;
- retry last proactive command within current USAT Interpreter navigation unit (the command which generated the current general result).

For system actions the attribute of the action TLV shall be ignored by the USAT Interpreter.

A navigation action is indicated by a service given action ID and one of the following USAT Interpreter data structures as "action to be performed":

- page reference TLV;
- anchor reference TLV.

For navigation actions the attribute of the action TLV shall be ignored by the USAT Interpreter.

A single USAT Interpreter byte code to be executed is indicated by a service given action ID and one of the following USAT Interpreter byte codes as "action to be performed":

- Display Text;
- Get Input;
- Set Variable;
- Execute USAT Command;
- Execute Native Command.

The attribute of the action TLV determines the behavior of the USAT Interpreter after execution of the single USAT Interpreter byte code.

[Summary of action management in Terminal Response Handler mechanism:](#)

Action to be performed			
	Action ID	used TLV	attribute handling
System actions			
process next byte code	'00'	none	attribute byte shall be ignored
quit USAT Interpreter without user confirmation	'01'	none	
go back one entry in history list	'02'	none	
retry last proactive command within current USAT Interpreter navigation unit	'03'	none	
RFU system actions	'04' to '1F'	RFU	
Navigation actions			
branch to another page	defined by service ('20' to 'FF')	page reference TLV	attribute byte shall be ignored
branch to another navigation unit		or anchor reference TLV	
Single USAT Interpreter byte codes			
Execute Native Command byte code	defined by service ('20' to 'FF')	Execute Native Command byte code TLV	behavior after execution of a single USAT Interpreter byte code as "action to be performed": <ul style="list-style-type: none"> - execute next byte code within current USAT Interpreter navigation unit - execute current proactive command within current USAT Interpreter navigation unit again
Execute Display Text byte code		Display Text byte code TLV	
Execute Set Variable byte code		Set Variable byte code TLV	
Execute Get Input byte code		Get Input byte code TLV	
Execute USAT Command byte code		Execute USAT Command byte code TLV	

NOTE: The retry action should be used only in conjunction with other actions or a notification text for a general result range to avoid the immediate repetition of the USAT command causing retry (possible senseless loop).

7.1.8.4.4 Action description

In the case of several actions (action list) assigned to the same general result range, a SELECT ITEM command shall be constructed by the USAT Interpreter using the corresponding action descriptions as items.

This TLV is mandatory if the modification type within the attribute byte of the terminal response handler modifier indicates "Replace" or "Add / Append".

If only one action is defined for the general result range, the action is executed by the USAT Interpreter without building the SELECT ITEM command.

After this SELECT ITEM command has been issued by the USAT Interpreter, an action shall be performed depending on the general result of the SELECT ITEM command itself:

General result for the SELECT ITEM	Comment
'00'...'0F' (ok)	the action defined for the option selected by the user shall be performed
'11' (backward move requested)	execute current proactive command within current USAT Interpreter navigation unit again or return to the wait state if the wait state is currently active
'10' (quit)	quit USAT Interpreter without user confirmation
'12'...'1F'	quit USAT Interpreter without user confirmation
'20'...'2F' (worth to retry)	quit USAT Interpreter without user confirmation
'30'...'3F' (not worth to retry)	quit USAT Interpreter without user confirmation

The parameters for the SELECT ITEM command shall be as follows:

- Alpha identifier not used;
- The command qualifier to be used for the SELECT ITEM command shall be '03' (["presentation type is specified in bit 2"](#) and ["presentation as a choice of navigation options"](#)).

7.2.2 Anchor

The content of this TLV is a sequence of bytes identifying the navigation unit. It is mandatory to provide this TLV, if a navigation unit of the current page or another page needs to branch to this navigation unit.

Coding:

Length	Value	Description	M/O
1	'0B'	Anchor Tag	M
1-3	L	Length	M
L	Data	Unique identification of navigation unit within the page. A sequence of bytes to uniquely identify the Anchor. This identification shall not contain a "#_"-character (coded '23') and is coded by the external system entity.	M

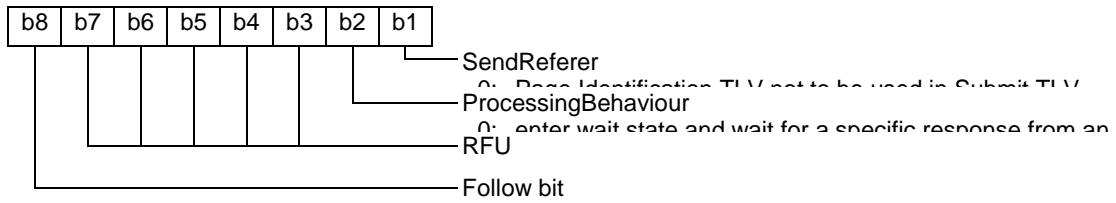
7.3 Anchor Reference

This TLV is used to refer to a navigation unit in the current page or in another page.

Length	Value	Description	M/O
1	'0C'	Anchor Reference Tag	M
1-3	L	Length	M
L	Data	Anchor Reference Name	M

An anchor reference name is the value part of a page identification TLV (unique identification of the page, see clause 7.1.2) followed by a '23' ("#") and the value part of the anchor TLV (unique identification of navigation unit, see clause 7.2.2) within the page. Either the page identification part or the anchor part (but not both) can be omitted. If the page identification part is omitted the reference is to an anchor on the current page. If the anchor name part is omitted the reference is to the ~~beginning~~ [first navigation unit](#) of the referenced page.

7.9.3.1 Attributes



If the SendReferer attribute is set, the Page Identification TLV of the current page shall be incorporated into the generated Submit TLV prior to the transmission to the external system entity.

If the ProcessingBehaviour attribute is not set the USAT Interpreter shall enter the wait state (see clause 4.2.2) after transmission.

If the ProcessingBehaviour attribute is set the USAT Interpreter shall render the next byte code after transmission.

7.10 Submit

This TLV is used to provide information from the USAT Interpreter to the external system entity. It shall be used only in the direction from the USAT Interpreter to the external system entity.

Length	Value	Description	M/O/C
1	'16'	Submit Tag	M
1-3	A+B	Length	M
A	TLV	Submit Data TLV	M
B	TLV	Page Identification TLV (if indicated in attribute "sendReferer" of Submit Configuration TLV)	C

7.10.1 Submit Data

The submit data information is a sequence of bytes ~~with substituted variables if contained~~. The origin of this TLV is the Submit Data TLV in the Submit Configuration TLV with variables substituted according to clause 6.3 method 2.

Length	Value	Description	M/O
1	'14'	Submit Data Tag	M
1-3	A	Length	M
A	Data	Byte sequence, according to clause 6.3 containing substituted variable references	O

7.10.2 Page Identification

This TLV shall be available if and only if the SendReferer bit in the attributes of the Submit Configuration TLV was set. It contains the page identification of the current page.

8.2.3 Ordered TLV List

One or more of this TLVs shall be contained in the "Assign and Branch" byte code.

This TLV encapsulates the

- "Inline Value 2", containing the text of a single item of the SELECT ITEM command;
- "Inline Value", containing a value to be assigned to the destination variable, if the item is selected; and
- "Page Reference", containing a destination for a branch, if the item is selected.

TLVs in the given order, which determine the action to be performed.

General variable assignments and navigation operations may be performed by the "Assign and Branch" byte code dependent on the data provided in the Ordered TLV List TLV. When optional TLVs are omitted, special cases can be encoded according to the following table:

Inline Value 2	Inline value (to be assigned to destination variable)	Page Reference	
present	present	present	"Display, Assign and Branch" Generation of a SELECT ITEM command by the USAT Interpreter. When the user has selected this item (described by the Inline Value 2 TLV) from the list, the USAT Interpreter shall assign the value of the Inline value TLV to the destination variable and branch to the navigation unit specified within the Page Reference TLV.
present	present	not present	"Set Variable Selected" Generation of a SELECT ITEM command by the USAT Interpreter. When the user has selected this item (described by the Inline Value 2 TLV) from the list, the USAT Interpreter shall assign the value of the Inline Value TLV to the destination variable and process next byte code .
present	not present	present	"Go Selected" Generation of a SELECT ITEM command by the USAT Interpreter. When the user has selected this item (described by the Inline Value 2 TLV) from the list, the USAT Interpreter shall branch to the navigation unit specified within the Page reference TLV. A destination variable identifier shall be ignored for this case.
present	not present	not present	"Select Item Display and Process next byte code" Generation of a SELECT ITEM command by the USAT Interpreter. When the user has selected this item (described by the Inline Value 2 TLV) from the list, the USAT Interpreter shall process the next byte code. A destination variable identifier shall be ignored for this case.
not present (see NOTE)	present	present	"Assign and Branch" No generation of a SELECT ITEM command by the USAT Interpreter. The USAT Interpreter shall assign the value of the Inline Value TLV to the destination variable and branch to the navigation unit specified within the Page reference TLV.
not present (see NOTE)	present	not present	"Set Variable" No generation of a SELECT ITEM command by the USAT Interpreter. The USAT Interpreter shall assign the value of the Inline value TLV to the destination variable.
not present (see NOTE)	not present	present	"Direct Go" No generation of a SELECT ITEM command by the USAT Interpreter. The USAT Interpreter shall directly branch to the navigation unit specified within the Page reference TLV. The destination variable identifier shall be ignored for this case.
not present	not present	not present	not valid, if occurs an error shall be issued.
<p>NOTE : - If Ordered TLVs containing an Inline Value 2 are present in the same Assign and Branch TLV, the USAT Interpreter shall ignore the Ordered TLVs which do NOT contain the Inline Value 2 TLV.</p> <p>- If Ordered TLVs not containing an Inline Value 2 are present in the same Assign and Branch TLV, the USAT Interpreter shall take into account the first Ordered TLV only.</p>			

8.7.4 Sequence of Simple TLVs and Simple TLV Indicators

The sequence of these Simple TLVs and Simple TLV Indicators is translated by the USAT Interpreter to form the sequence of Simple TLVs of an USAT command (3GPP TS 31.111 [1]). When expanding Simple TLV Indicators to Simple TLVs the length of the BER-TLV of the resulting USAT command shall be adjusted by the USAT Interpreter before issuing the command to the UE.

When executing a Execute USAT command byte code, the USAT Interpreter issues a regular USAT command to the UE using the USAT protocol. The translation procedure from the Execute USAT Command TLV to an USAT command can be visualised in principle as follows:

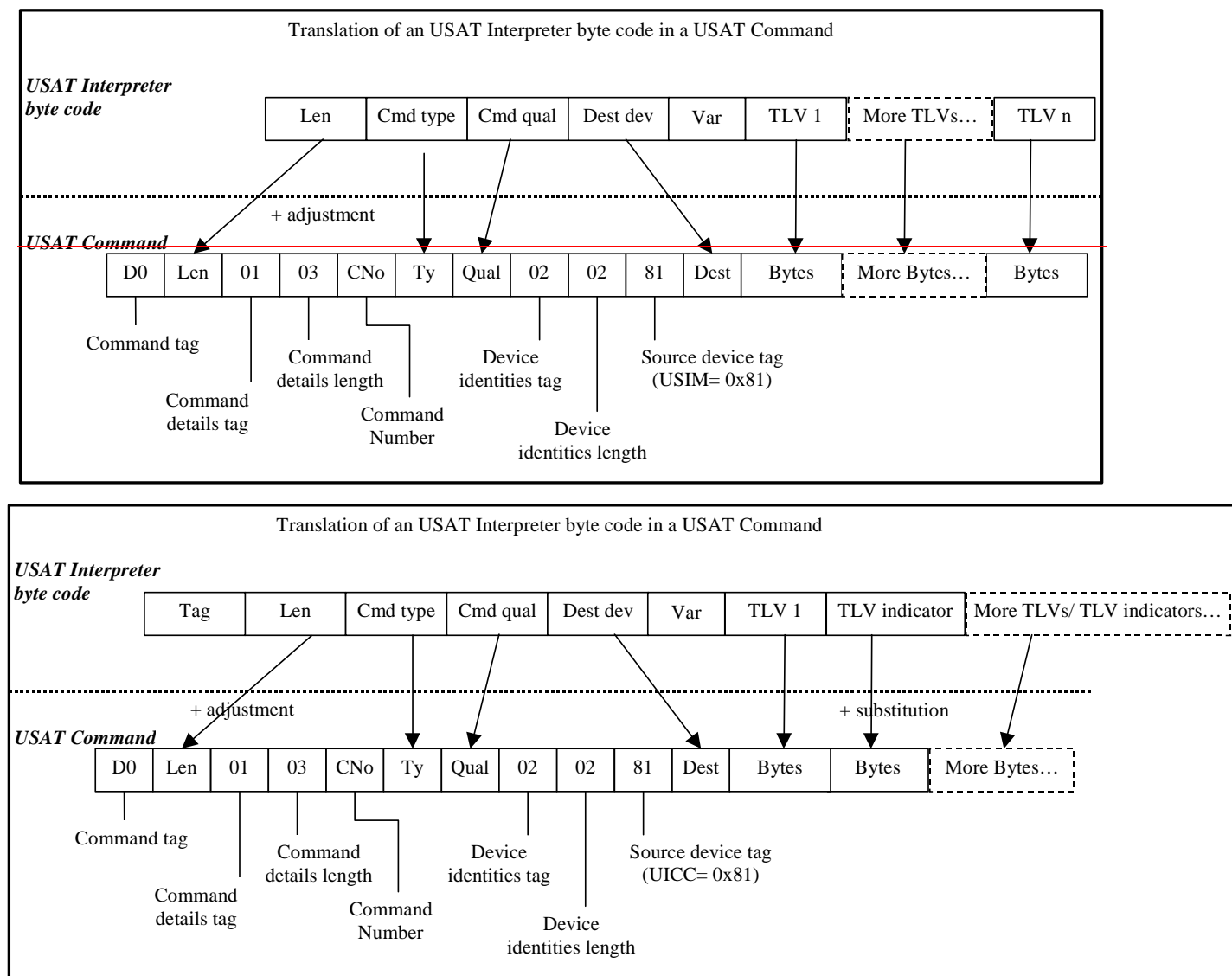


Figure 8.1

8.7.5.2 Optimisation Required

Only the first TLVs after the Result Simple TLV within a Terminal Response (see 3GPP TS 31.111 [1]) shall be processed by the USAT Interpreter as follows:

- If the first TLV after the Result Simple TLV is a Text String TLV according to 3GPP TS 31.111 [1], the value part without the DCS byte is assigned to the result variable. The DCS is removed from the V field, but used for variable management internally by the USAT Interpreter.
- In all other cases, the value part of the first TLV after the Result Simple TLV is assigned to the result variable. The type "unknown" shall be used for variable management internally by the USAT Interpreter.

8.8.2 Result of a Native Function Call

When the native function call returns, the values produced by the call are stored ~~the values associated with~~ in the variables referenced by ~~in~~ the output list.

9 Native Commands

Native Commands or "plug-ins" shall be used to provide specific functionality not contained in the USAT Interpreter byte code set. This can be e.g. operating system calls, execution of specific security algorithms, calculation routines or conversion routines. All native commands are called using the Execute Native Command byte code.

Each native command shall have a Native Code Identifier. The Native Code Identifier has a size of 2 bytes and is binary coded, most significant byte first. The values '0000' to '7FFF' are RFU for native commands specified in the present document. Other values may be used for proprietary implementations.

Native Commands [defined below](#) are optionally to be supported by the USAT Interpreter. If [any of these](#) Native Commands are supported by the USAT Interpreter ~~(~~which are specified within the present document ~~)~~ (using a NCI specified in the present document), they shall be implemented according to the present document.

Native commands specified by the present document:

This is for further study.

CR-Form-v3

CHANGE REQUEST

⌘ **31.113 CR 003** ⌘ rev **-** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Changes to USAT Interpreter system information partition table		
Source:	⌘ T3		
Work item code:	⌘ USAT Interpreter	Date:	⌘ 06-11-2001
Category:	⌘ C	Release:	⌘ REL-5
	Use <u>one</u> of the following categories: F (essential correction) A (corresponds to a correction in an earlier release) B (Addition of feature), C (Functional modification of feature) D (Editorial modification)		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)
	Detailed explanations of the above categories can be found in 3GPP TR 21.900.		

Reason for change:	⌘ Split of "USAT Interpreter profile" list into USAT commands and functionality provided by the USAT Interpreter.
Summary of change:	⌘ Changes to the table in 6.1.1.1 were applied: "USAT Interpreter profile" has been renamed to "USAT Command Filter"; a new variable called "USAT Interpreter Byte Code Filter" has been introduced.
Consequences if not approved:	⌘ Commands and functionality can not be handled (administered) separately by card issuer.

Clauses affected:	⌘ 6.1.1.1
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications ⌘ <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications
Other comments:	⌘

6.1.1.1 USAT Interpreter system information partition

The USAT Interpreter partition is preloaded during the manufacturing process of the USIM or during the runtime of the USAT Interpreter.

At least the following information shall be stored:

Variable ID	Description	Coding
'00'	ICCID of UICC	Binary coding as for EF _{ICCID} specified in SCP TS 102 221 [4]
'01'	USAT Interpreter version	<p>Byte 1: Issuer Version USAT Interpreter issuer specific version. The coding and value of this byte depends on the USAT Interpreter issuer. The USAT Interpreter issuer is stored in variable '07' and variable '08'.</p> <p>Bytes 2-3: TS 31.113, Version (this TS) Byte 2: first digit (x according to the foreword of the present document) of the version of the supported 3GPP TS 31.113; BCD coded</p> <p>Byte 3: second digit (y according to the foreword of the present document) of the version of the supported 3GPP TS 31.113; BCD coded</p> <p>Bytes 4-5: Version of TS 31.114 [2] Byte 2: first digit (x according to the foreword of the present document) of the version of the supported 3GPP TS 31.114; BCD coded</p> <p>Byte 3: second digit (y according to the foreword of the present document) of the version of the supported 3GPP TS 31.114; BCD coded</p> <p>further bytes are RFU</p> <p>Example: Issuer version: '22' TS 31.113 version: 5.2.0 TS 31.114 version: 5.12.3 resulting coding: '22 05 02 05 12'</p>
'02'	USAT Command Filter Interpreter profile (features)	<p>According to administrative configuration coding as specified in TS 31.114 [2]. This includes the list of supported and forbidden USAT Commands. This includes the list of allowed USAT Commands. Coding as specified in TS 31.114 [2].</p> <p><u>NOTE: Content is dynamic, i.e. it is impacted by the current configuration</u></p>
'03'	USAT Interpreter Native Commands	List of supported native commands. Coding: Sequence of NCIs. Each NCI coded in 2 bytes.
'04'	Terminal Profile as got at runtime	Binary coded as defined in 3GPP TS 31.111 [1] for TERMINAL PROFILE
'05'	Error Code as generated by the last byte code command executed	Binary coded as specified in clause 12
'06'	Maximum page size for temporary storage of one page	Binary coded, number of bytes available for page storage, coded in 2 bytes indicating the number of blocks of memory consisting of 128 bytes each, most significant byte first
'07'	USAT Interpreter issuer identification	URL of USAT Interpreter issuer, coding according to RFC 1738 [9] <host> of URL.
'08'	Hash Value of URL of USAT Interpreter issuer identification	4 most significant (left most) bytes of SHA-1 hash of the content of variable '07'
'09'	Buffer Sizes	<p>First 2 bytes, binary coded, MSB first:</p> <ul style="list-style-type: none"> Receive buffer size in bytes available for messages to be received by the USAT Interpreter. <p>Second 2 bytes, binary coded, MSB first:</p> <ul style="list-style-type: none"> Transmit buffer size in bytes available for

		<p>messages to be sent by the USAT Interpreter.</p> <p>Both sizes include all possibly needed space for transport headers, security, routing information, concatenation information and so on.</p>
'0A'	USAT Interpreter Byte Code Filter	<p>This includes the list of allowed USAT Interpreter byte codes.</p> <p>Coding as specified in TS 31.114 [2].</p> <p>NOTE: Content is dynamic, i.e. it is impacted by the current configuration.</p>
'0BA'... '13'	RFU	

CHANGE REQUEST

⌘ **31.113 CR 004** ⌘ rev ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ USAT interpreter : comparison with a variable value		
Source:	⌘ T3		
Work item code:	⌘ USAT interpreter	Date:	⌘ 07/11/2001
Category:	⌘ B	Release:	⌘ Rel-5
	<p>Use <u>one</u> of the following categories:</p> <p>F (essential correction) A (corresponds to a correction in an earlier release) B (Addition of feature), C (Functional modification of feature) D (Editorial modification)</p> <p>Detailed explanations of the above categories can be found in 3GPP TR 21.900.</p>		<p>Use <u>one</u> of the following releases:</p> <p>2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)</p>

Reason for change:	⌘ The USAT Interpreter byte code specification doesn't include the "Inline Value" tag in the list of elements allowed in the "BranchOnVariableValue" byte code. That means if a service developer want to branch to other locations depending on a variable value, he must create other variables containing the values to be tested. In addition to be an inconvenient constraint for the service developer, this will have a big impact on the USAT Interpreter memory usage: the constants used for comparison will not stay in the page itself, they will be duplicated in the USAT Interpreter variable memory.
Summary of change:	⌘ Just allow also the "Inline value" Element to be used as a value to be compared against the evaluated variable.
Consequences if not approved:	⌘ - Design constraint for the services (and/or the USAT Gateway) - Inefficient memory management

Clauses affected:	⌘ 8, 8.5 , 8.5.2	
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications ⌘ <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	
Other comments:	⌘ 	

8 USAT Interpreter byte codes

Each USAT Interpreter byte code is a TLV. Each byte code has its own byte code tag value, optional attributes and a list of arguments. Arguments, if present, shall appear in the order given.

The byte codes make use of the USAT Interpreter TLVs as follows:

	Attribute Bytes	Variable References	Variable Identifier List TLV	Inline Value TLV	Inline Value 2 TLV	Page Reference TLV	Ordered TLV List TLV	Input List TLV
Set Variable		✓		✓				
Assign and Branch		✓		✓	✓	✓	✓	
Extract		✓						
Go Back	1							
Branch on Variable Value		✓	✓	✓		✓	✓	
Exit	1		✓					
Execute USAT Command	1	✓						
Execute Native Command	1		✓					✓
Get Length		✓	✓					
Get TLV Value		✓	✓					
Display Text	1			✓				
Get Input	1	✓		✓	✓			

8.5 Branch On Variable Value

This byte code compares a variable to a list of values that have an associated Page Reference. When a match is found, the referenced page shall be executed. If no match is found, the first Page Reference after the Ordered TLV List shall be used to branch. If this last Page Reference TLV is not contained in the byte code, no branch shall be executed and the USAT Interpreter shall continue to render the next byte code after the Branch on Variable Value byte code.

Length	Value	Description	M/O
1	'44'	Branch on Variable Value Tag	M
1	1+A+...+X+Y	Length	M
1	Data	Variable ID (containing the value to match)	M
A	TLV	Ordered TLV List TLV (see description below) containing: <ul style="list-style-type: none"> - Variable Identifier List TLV (referring to one variable containing the value to be compared with the match value, additional Variable IDs to be ignored) or Inline Value TLV - Page Reference TLV, to branch to, if value matches 	M
...	
X	TLV	Ordered TLV List TLV	O
Y	TLV	Page Reference TLV, if no match is found, go to this reference	O

Possible errors:

Error Code	Description	Action
No error	OK	Continue
Reference to undefined	Reference to undefined variable	Stop
Jump to undefined	Page Reference not found.	Stop

Explanation of used arguments:

8.5.1 Variable ID

This variable shall contain the value to be compared.

8.5.2 Ordered TLV List

In each of these TLVs the following TLVs are encapsulated:

- Variable Identifier List TLV (referring to one variable containing the value to be compared with the match value; additional Variable IDs to be ignored)

OR

[Inline Value TLV \(directly containing the value to be compared with the match value\)](#)

- Page Reference TLV.

The Page Reference TLV contains the location to be branched to, if the comparison is successful.