


# S@T 01.20 v4.0.0 (Release 2009)

S@T Session Protocol

Published by  **simalliance** now Trusted Connectivity Alliance

Copyright © 2009 Trusted Connectivity Alliance Ltd



# S@T 01.20 v4.0.0 (Release 2009)

---

*Technical Specification*



## **S@T Session Protocol**



## TABLE OF CONTENTS

<b>1</b>	<b>Terminology</b> .....	<b>4</b>
1.1	Abbreviation.....	4
<b>2</b>	<b>LIST OF DOCUMENTS</b> .....	<b>4</b>
<b>3</b>	<b>OVERVIEW</b> .....	<b>6</b>
<b>4</b>	<b>SSP PROTOCOL</b> .....	<b>6</b>
4.1	Command usage summary.....	7
4.2	Structure of SSP packet .....	8
<b>5</b>	<b>Structure of SSP command</b> .....	<b>9</b>
5.1	Coding of SSP command type .....	9
5.2	SSP parameters .....	9
5.2.1	Session Id.....	9
5.2.2	Connection Id.....	10
5.2.3	Protocol Id .....	10
5.2.4	Transaction Id .....	10
5.2.5	Connection Status .....	10
5.2.6	Gateway/Application address.....	11
<b>6</b>	<b>SSP Commands</b> .....	<b>12</b>
6.1	CONNECT_REQ.....	12
6.2	CONNECT_RSP .....	12
6.3	DISCONNECT_REQ .....	13
6.4	DATA_REQ.....	14
6.5	DATA_RSP.....	14
6.6	GET_REQ .....	15
6.7	POST_REQ .....	15
6.8	REPLY_RSP .....	16
6.9	EXPRESS_DATA_REQ.....	17
6.9.1	Browser Info .....	17
<b>7</b>	<b>IMPLEMENTATION REQUIREMENTS</b> .....	<b>18</b>
7.1	Reception of CONNECT_REQ / Transmission of CONNECT_RSP. ....	18
7.2	Reception of CONNECT_RSP .....	18
7.3	Reception of DATA_REQ / Transmission of DATA_RSP.....	19
7.4	Reception of DATA_RSP .....	19
7.5	Reception of GET_REQ/POST_REQ / Transmission of REPLY_RSP .....	19
7.6	Reception of REPLY_RSP.....	20
7.7	Reception of EXPRESS_DATA_REQ .....	20
7.8	Reception of DISCONNECT_REQ.....	20



**8 HISTORY..... 22**

**8.1 Annex A [informative] : List of CRs.....23**



# 1 Terminology

## 1.1 Notation

Prefix '0x' indicates hexadecimal value.

## 1.2 Abbreviation

HTTP	Hyper Text Transfer Protocol
MSB	Most Significant Bit
S@T	SIM Alliance Toolbox
SSP	S@T Session Protocol
TLV	Tag Length Value encoding

## 1.3 Definitions

**Administration protocol:** based on SSP and used for transmitting administrative commands as defined in /Admin/

**Connected state:** state of the browser/gateway when a session has been established

**Connecting state:** state of the browser/gateway is waiting for the session establishment (i.e. it has sent CONNECT\_REQ and is waiting for CONNECT\_RSP).

**Disconnected state:** no SSP session is opened

**High priority push protocol:** based on SSP and used for transmitting push commands as defined in /Push/

**Idle state:** state of the browser when no proactive session is running (no proactive command is pending).

**Pull protocol:** based on SSP and used for transmitting operational commands as defined in /Pull/

**S@T Session protocols:** basic protocol for S@T browser/gateway communication based on the session establishment and data exchange within the established session.

**Waiting for response state:** state of the browser when it keeps the proactive session alive by issuing DISPLAY TEXT to notify the user that the browser is waiting for or receiving the gateway response.

# 2 LIST OF DOCUMENTS

/23.040/	3GPP TS 23.040: "Technical realization of Short Message Service (SMS)".
/23.048/	3GPP TS 23.048: "Security mechanisms for the (U)SIM application toolkit; Stage 2".
/SBC/	S@T 01.21: "S@T Bytecode".
/Admin/	S@T 01.21: "S@T Administrative Commands".
/Pull/	S@T 01.22: "S@T Operational Commands".
/Push/	S@T 01.23: "S@T Push Commands".



This document is part of a specification set, please refer to "S@T Release Note" for a comprehensive document list, including document versions.



### 3 OVERVIEW

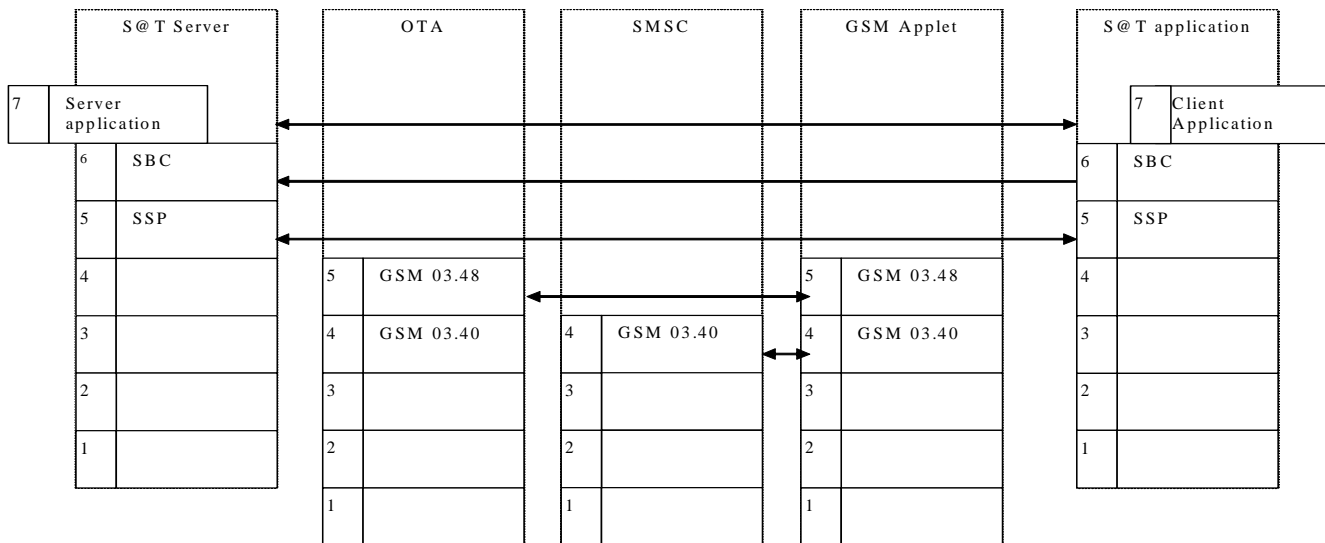


Fig 1. Layers Overview.

The way to present layers used in this document assumes that each layer will be fully independent from the layer above. The SSP layer protocol can be used on any connected or non connected media, and guarantees that upper layers are bearer independent by allowing the same functionalities regardless of the media used.

The SSP layer will be used to connect a S@T browser to a S@T gateway (via a SMS-C and an OTA Entity for example) in order to send and receive data.

The SSP layer provides an efficient way to exchange messages and to prevent delayed messages from disrupting an implementation using SMS.

### 4 SSP PROTOCOL

The coding is specified for a set of session commands that allow the management of a session layer between a S@T browser and a S@T gateway.

The special TAR value '534054' ("S@T") shall be used for the 23.048 message header containing SSP packet.



## 4.1 Command usage summary

A session command is issued in a XXX\_REQ or XXX\_RSP form. Upon reception of XXX\_REQ a S@T browser/gateway normally responds with XXX\_RSP according to the following table:

<i>XXX_REQ</i>	<i>XXX_RSP</i>	<i>Description</i>
CONNECT_REQ	CONNECT_RSP	Establishment of a session (connection).
DATA_REQ	DATA_RSP	Regular data transfer.
GET_REQ	REPLY_RSP	Special data request with parameters.
POST_REQ	REPLY_RSP	Special data request with parameters.
EXPRESS_DATA_REQ	Not present	S@T browser information.
DISCONNECT_REQ	Not present	Session release (disconnection).

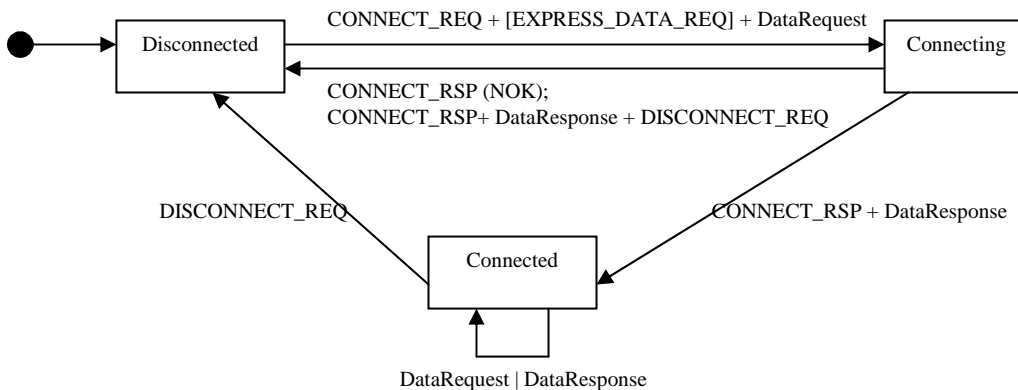




## 4.2 Structure of SSP packet

SSP packet consists of one or more SSP commands with the same Session Id. Structure of SSP packet is protocol dependent and defined in the protocol specifications (see possible S@T protocols in 5.2.3).

In general, SSP packet exchange can be presented with the following high level state diagram:



where:

- DataRequest: one or more DATA\_REQ commands or GET\_REQ or POST\_REQ;
- DataResponse: one or more DATA\_RSP commands or REPLY\_RSP.

When an entity (S@T browser or S@T gateway) is disconnected it shall send CONNECT\_REQ command followed by DataRequest with Session Id = 0x00 and go to the connecting state.

An entity goes from the connecting state to the connected state on receiving successful CONNECT\_RSP followed by DataResponse with the established Session Id. If the SSP package is followed by DISCONNECT\_REQ, then the entity goes to the disconnected state after processing the response data. If session cannot be established, the entity will receive single CONNECT\_RSP command with unsuccessful Connection Status and shall go to the disconnected state.

When an entity is connected, it can send DataRequest and get DataResponse with the current Session Id. If the entity receives DISCONNECT\_REQ on DataRequest, it shall go to the disconnected state.

NOTE: the S@T gateway may also go to the disconnected state on timeout.

NOTE: the S@T browser shall also go to the disconnected state when it exits (i.e. goes to the idle state).



## 5 Structure of SSP command

A SSP command consists of command type, one or more SSP parameters and optional command data (protocol specific):

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	V	Command type	M
A	V	SSP parameters	M
1 - 3	B	Length of command data (length coded in BER-TLV)	O
B	V	Command data (protocol specific)	O

### 5.1 Coding of SSP command type

SSP command types are coded with the following values:

<i>SSP COMMAND</i>	<i>CODE</i>
CONNECT_REQ	0x01
CONNECT_RSP	0x10
DISCONNECT_REQ	0x05
DATA_REQ	0x08
DATA_RSP	0x80
GET_REQ	0x02
POST_REQ	0x04
REPLY_RSP	0x20
EXPRESS_DATA_REQ	0x06

### 5.2 SSP parameters

#### 5.2.1 Session Id

A Session Id is a number identifying an unique session between the S@T browser and the S@T gateway. The Session Id is initialized by the entity on CONNECT\_REQ reception.

The Session Id is coded on 1 byte, the MSB indicates the direction of the CONNECT\_RSP command. If CONNECT\_RSP is sent by the S@T browser, the first bit of the Session Id is set to 1; If CONNECT\_RSP is sent by the S@T gateway the first bit of the session Id is set to 0. Special value 0x00 is reserved for not established Session Id.

Session Id initialized by the *S@T browser*:

7 6 5 4 3 2 1 0

---



1	X	X	X	X	X	X	X
---	---	---	---	---	---	---	---

Session Id initialized by the S@T gateway (except 0x00):

7 6 5 4 3 2 1 0

0	X	X	X	X	X	X	X
---	---	---	---	---	---	---	---

The Session Id is allocated to a S@T gateway / S@T browser couple and MUST remain valid and reserved until S@T browser/gateway goes to the disconnected state.

Simultaneous session establishment is not allowed.

## 5.2.2 Connection Id

Connection Id is coded on 1 byte. It is used to identify messages exchanged during connecting (session establishment).

## 5.2.3 Protocol Id

Protocol Id is coded on 1 byte. It is used to enable SSP command data to be typed, and allow above layers to handle correctly this encapsulated data. When establishing a connection, the protocol needing to exchange information in SSP messages must be specified. A list of possible Protocol Id values is:

<i>Protocol Id</i>	<i>Protocol Name</i>
0x01	Pull (see /Pull/)
0x02	Administration (see/Admin/)
0x03	High priority push (see /Push/)

NOTE: Other Protocol Id values are RFU. CONNECT\_REQ with unknown Protocol Id and all the following commands in the same SSP message shall be ignored.

## 5.2.4 Transaction Id

Transaction Id is coded on 1 byte. The role of the Transaction Id is to identify a message.

Management of the Transaction Id is application dependent. During a session it shall be incremented at each data request command sent (DATA\_REQ, POST\_REQ, GET\_REQ).

## 5.2.5 Connection Status

Connection Status is coded on 1 byte. The role of the Connection Status is to indicate a successful session allocation result or the reason why a new session cannot be allocated or an existing session shall be closed.

Possible values for this field are:

<i>Connection Status</i>	<i>CODE</i>
NO_ERROR	0x00
COMMAND_SYNTAX_ERROR	0x81



SERVER_NOT_FOUND	0x82
SERVER_BUSY	0x83
CONNECTION_REFUSED	0x84
TIME_OUT	0x85
TOO_MANY_SESSIONS	0x86
SESSION_ERROR	0x88
NOT_SPECIFIED	0x8F

NOTE: All other values are RFU and shall be treated as NOT\_SPECIFIED.

### 5.2.6 Gateway/Application address

Gateway/Application address is coded on 6 bytes:

- Gateway address: 3 bytes; defines the targeted S@T gateway (defines the default S@T gateway if set to zeroes);
- Application address: 3 bytes; defines the originating application. (indicates default S@T Application if set to zeroes)

It is recommended to code gateway/application address in 000 000.



## 6 SSP Commands

### 6.1 CONNECT\_REQ

The CONNECT\_REQ command is used by the S@T Browser/gateway to ask the S@T gateway/Browser to open a session.

For optimization purposes, protocol specific commands (EXPRESS\_DATA\_REQ, GET\_REQ, POST\_REQ, DATA\_REQ) shall follow a CONNECT\_REQ command in the same message. In this case, the receiver MUST ignore the Session Id parameter of the commands following the CONNECT\_REQ command if it is equal to 0x00 and consider such commands as being part of the same session.

Commands other than EXPRESS\_DATA\_REQ, GET\_REQ, POST\_REQ, DATA\_REQ must not follow a CONNECT\_REQ in the same message.

CONNECT\_REQ coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x01	CONNECT_REQ command type	M
1	according to section 5.2.3	Protocol Id	M
1	according to section 5.2.2	Connection Id	M
6	according to section 5.2.6	Gateway/Application address	M

### 6.2 CONNECT\_RSP

The CONNECT\_RSP command is used by an application to respond to a connection request (CONNECT\_REQ). There is no acknowledgement of this message.

The entity receiving a CONNECT\_REQ and responding with a CONNECT\_RSP is in charge of allocating a Session Id. If Connection Status is successful, the connection initiator will get this Session Id and each entity will insert this number in following messages.

If connection failed to achieve, each entity MUST return to the “Disconnected” state.

If Connection Status is successful, for optimization purposes, the other SSP commands shall follow CONNECT\_RSP in the same message. Commands other than DATA\_RSP, REPLY\_RSP, DISCONNECT\_REQ must not follow a CONNECT\_RSP in the same message.

CONNECT\_RSP coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x10	CONNECT_RSP command type	M
1	the same Connection Id as in CONNECT_REQ	Connection Id	M
1	according to section 5.2.1	Session Id	M
1	according to section 5.2.5	Connection Status	M



## 6.3 DISCONNECT\_REQ

The DISCONNECT\_REQ command is used by the S@T Gateway or the S@T Browser to stop an established session (disconnect). There is no acknowledgement of this message. Furthermore, this message is NOT MANDATORY to terminate a session. Implementations MAY use time out to discard unused sessions. The S@T browser also goes to the “Disconnected” state on exit.

Other SSP commands shall not follow this command in the same message.

DISCONNECT\_REQ coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x05	DISCONNECT_REQ command type	M
1	according to section 5.2.1	The current Session Id	M
1	according to section 5.2.5	Connection Status	M



## 6.4 DATA\_REQ

The DATA\_REQ command is used to exchange regular data.

For optimization purposes, other DATA\_REQ commands MAY follow this command.

DATA\_REQ coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x08	DATA_REQ command type	M
1	according to section 5.2.1	The current Session Id	M
1	according to section 5.2.4	Transaction Id	M
1 - 3	B	Length of command data (length coded in BER-TLV)	M
B	V	Command data (protocol specific)	M

## 6.5 DATA\_RSP

The DATA\_RSP command is used to exchange regular data, it is sent in response to a received DATA\_REQ.

For optimization purposes, other DATA\_RSP commands MAY follow this command. DISCONNECT\_REQ command may follow DATA\_RSP commands.

DATA\_RSP coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x80	DATA_RSP command type	M
1	according to section 5.2.1	The current Session Id	M
1	the same Transaction Id as in DATA_REQ	Transaction Id	M
1 - 3	B	Length of command data (length coded in BER-TLV)	M
B	V	Command data (protocol specific)	M



## 6.6 GET\_REQ

The GET\_REQ command is to implement the "GET" as specified in RFC 2068 (Hypertext Transfer Protocol -- HTTP/1.1).

No other SSP commands shall follow the GET\_REQ in the same SSP packet.

GET\_REQ coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x02	GET_REQ command type	M
1	according to section 5.2.1	The current Session Id	M
1	according to section 5.2.4	Transaction Id	M
1 - 3	B	Length of command data (length coded in BER-TLV)	M
B	V	Command data (protocol specific)	M

## 6.7 POST\_REQ

The POST\_REQ command is to implement the "POST" as specified in RFC 2068 (Hypertext Transfer Protocol -- HTTP/1.1).

No other SSP commands shall follow the POST\_REQ in the same SSP packet.

POST\_REQ coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x04	POST_REQ command type	M
1	according to section 5.2.1	The current Session Id	M
1	according to section 5.2.4	Transaction Id	M
1 - 3	B	Length of command data (length coded in BER-TLV)	M
B	V	Command data (protocol specific)	M





## 6.8 REPLY\_RSP

The REPLY\_RSP command MUST be used to reply to a POST\_REQ or a GET\_REQ command.

No other commands shall follow the REPLY\_RSP in the same SSP packet.

REPLY\_RSP coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x20	REPLY_RSP command type	M
1	according to section 5.2.1	The current Session Id	M
1	the same Transaction Id as in GET_REQ or POST_REQ	Transaction Id	M
1 - 3	B	Length of command data (length coded in BER-TLV)	M
B	V	Command data (protocol specific)	M



## 6.9 EXPRESS\_DATA\_REQ

The EXPRESS\_DATA contains S@T browser information. The EXPRESS\_DATA command MUST be sent by the S@T browser commands just after the CONNECT\_REQ command.

For optimization purposes, protocol specific commands (GET\_REQ, POST\_REQ, DATA\_REQ) shall follow an EXPRESS\_DATA\_REQ command in the same message.

EXPRESS\_DATA\_REQ coding:

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x06	EXPRESS_DATA_REQ command type	M
1	according to section 5.2.1	The current Session Id	M
1 - 3	A + B1 + .. + Bn	Length of command data (length coded in BER-TLV)	M
A	Browser Info TLV (see 6.9.1 )	Browser info	M
B1	TLV	Environment variable 1	O
...	...	...	O
Bn	TLV	Environment variable N	O

NOTE: Environment variable is coded as TLV, where tag is environment variable Id and value is environment variable value as defined in /SBC/. It shall be possible to configure environment variable TLVs in EXPRESS\_DATA\_REQ at the personalisation stage.

### 6.9.1 Browser Info

This TLV contains a S@T browser information that is sent to a S@T gateway to adapt to different S@T browser types

<i>LENGTH</i>	<i>VALUE</i>	<i>DESCRIPTION</i>	<i>M/O</i>
1	0x50	Browser info tag	M
1	4	Length	M
1	The same coding as for environment variable 'Browser Supplier' defined in /SBC/	Browser Supplier	M
1	The same coding as for environment variable 'Browser Version' defined in /SBC/	Browser Version	M
2	The same coding as for environment variable 'Reception Buffer Size' defined in /SBC/	Reception Buffer Size	M



## 7 IMPLEMENTATION REQUIREMENTS

This paragraph states implementation requirements. They do concern verifications that have to be made when receiving/sending a message.

### 7.1 Reception of CONNECT\_REQ / Transmission of CONNECT\_RSP.

Upon reception of a CONNECT\_REQ, the S@T browser/gateway verifies:

- if it can allocate a session;
- Protocol Id is known; otherwise the whole SSP packet shall be ignored silently.

If the entity (S@T browser/gateway) accepts a connection, it shall:

- go to the connected state;
- generate a CONNECT\_RSP with:
  - Connection Id : identical to the one received in the CONNECT\_REQ;
  - Session Id : set to a new allocated value. It will be used through the whole session;
  - Connection Status: set to NO\_ERROR;
- handle other data request commands follow the received CONNECT\_REQ in the same message. If their Session Ids are set to 0x00, they are considered to be the same as the newly allocated one;
- add data response commands in the response message after the CONNECT\_RSP.

Upon reception of a CONNECT\_REQ, the *S@T browser* shall send CONNECT\_RSP only if it can allocate a session. Otherwise the S@T browser shall ignore silently CONNECT\_REQ and all the following commands in the same SSP packet.

If the *S@T gateway* can not allocate a session :

- it MUST return a CONNECT\_RSP with:
  - Connection Id : Identical to the one received in the CONNECT\_REQ ;
  - Session Id : unset (0x00) ;
  - Connection Status: set to an error code;
- commands following CONNECT\_REQ in the same message MUST be ignored

### 7.2 Reception of CONNECT\_RSP

The S@T browser/gateway accepts CONNECT\_RSP if:

- it is in the connecting state; the S@T browser shall also be in the waiting for response state
- the Connection Id is equal to the one sent in the corresponding CONNECT\_REQ.



Otherwise the S@T browser/gateway shall ignore the whole SSP packet.

If CONNECT\_RSP is accepted then the S@T browser/gateway shall verify Connection Status:

- NO\_ERROR: the S@T browser/gateway gets the Session Id from the CONNECT\_RSP, goes into the connected state and handles other commands followed CONNECT\_RSP in the same packet.
- Error: the S@T browser/gateway goes to the disconnected state. Other commands followed CONNECT\_RSP in the same packet shall be ignored.

### 7.3 Reception of DATA\_REQ / Transmission of DATA\_RSP

The S@T browser/gateway accepts DATA\_REQ if:

- it is in the connected state;
- Session Id in DATA\_REQ is equal to the current Session Id
- DATA\_REQ data is coded properly (according to S@T protocol)

Otherwise the S@T browser/gateway shall ignore and skip the DATA\_REQ command.

If DATA\_REQ is accepted then the S@T browser/gateway shall return DATA\_RSP with:

- Session Id: is equal to the current Session Id;
- Transaction Id: is equal to the one received in the DATA\_REQ;
- Data: protocol dependent data.

If other commands follow this command in the same message, they MUST be handled.

### 7.4 Reception of DATA\_RSP

The S@T browser/gateway accepts DATA\_RSP if:

- it is in the connected state; the S@T browser shall also be in the waiting for response state
- Session Id in DATA\_RSP is equal to the current Session Id
- Transaction Id in DATA\_RSP is equal to the one sent in the corresponding DATA\_REQ.
- DATA\_RSP data is coded properly (according to S@T protocol )

Otherwise the S@T browser/gateway shall ignore and skip the DATA\_RSP command.

If other commands follow this command in the same message, they MUST be handled.

### 7.5 Reception of GET\_REQ/POST\_REQ / Transmission of REPLY\_RSP

The S@T browser/gateway accepts GET\_REQ/POST\_REQ if:

- it is in the connected state;
- Session Id in GET\_REQ/POST\_REQ is equal to the current Session Id



- GET\_REQ/POST\_REQ data is coded properly (according to S@T protocol )

Otherwise the S@T browser/gateway shall ignore and skip the GET\_REQ/POST\_REQ command.

If DATA\_REQ is accepted then the S@T browser/gateway shall return REPLY\_RSP with:

- Session Id: is equal to the current Session Id;
- Transaction Id: is equal to the one received in the GET\_REQ/POST\_REQ;
- Data: protocol dependent data.

If other commands follow this command in the same message, they MUST be handled.

## 7.6 Reception of REPLY\_RSP

The S@T browser/gateway accepts REPLY\_RSP if:

- it is in the connected state; the S@T browser shall also be in the waiting for response state;
- Session Id in REPLY\_RSP is equal to the current Session Id;
- Transaction Id in REPLY\_RSP is equal to the one sent in the corresponding GET\_REQ/POST\_REQ;
- REPLY\_RSP data is coded properly (according to S@T protocol).

Otherwise the S@T browser/gateway shall ignore and skip the REPLY\_RSP command.

If other commands follow this command in the same message, they MUST be handled.

## 7.7 Reception of EXPRESS\_DATA\_REQ

The S@T gateway accepts EXPRESS\_DATA\_REQ if:

- it is in the connected state;
- Session Id in REPLY\_RSP is equal to the current Session Id;
- EXPRESS\_DATA\_REQ data is coded properly.

Otherwise the S@T gateway shall ignore and skip the EXPRESS\_DATA\_REQ command.

The S@T browser shall ignore and skip EXPRESS\_DATA\_REQ command.

If other commands follow this command in the same message, they MUST be handled.

## 7.8 Reception of DISCONNECT\_REQ

The S@T browser/gateway accepts DISCONNECT\_REQ if:

- it is in the "Connected" state; the S@T browser shall also be in the waiting for response state;



- Session Id in DISCONNECT\_REQ is equal to the current Session Id.

If DISCONNECT\_RSP is accepted then the S@T browser/gateway shall go to the disconnected state regardless of the Connection Status value in DISCONNECT\_RSP.

If other commands follow this command in the same message, they MUST be ignored.



## 8 HISTORY

Document history		
Release	Approved by	Comment
V1.0.0	SIM Alliance TDG	First internal release
V1.0.1	SIM Alliance TDG	Update Layout
V1.0.2	SIM Alliance TDG	Final release
V1.0.3	SIM Alliance TDG	Example of using S@T Protocols added. Annex describing Optional features added
V1.0.4	SIM Alliance TDG	Working document
V1.0.5	SIM Alliance TDG	Editorial changes
V1.0.6	SIM Alliance TDG	Editorial changes for publication
V2.0.0	SIM Alliance <a href="#">S@T</a> Group	Clarifications on CR and editorial changes for publication June 2004 CR 2004-13 CR 2004-22 CR 2004-27
V3.0.0	SIM Alliance TDG	Editorial modifications Release 2007 publications
V3.0.1	SIM Alliance TDG	Section 14.9: PRISM-Dec-2007 #003
V4.0.0	SIM Alliance TDG	S@T Task Force: Review of Spec: Major editorial modifications for clarification <ul style="list-style-type: none"><li>- Definition of BrowserInfo sent in EXPRESS_DATA_REQ</li><li>- Clarification of implementation requirements</li><li>- Removed multi session management</li></ul>



## 8.1 Annex A [informative] : List of CRs

CR Number	CR Identifier	Subject	Document Reference	Status / Meeting No.
10105	SCHLUMBERGE R-WG1 -MAR-2001#1	Security LEVEL ERROR MANAGEMENT	S@T 1.20 V1.0.3	Accepted#27
10119	GEMPLUS-WG1-MAY-2001#28-2	Protocol Id declaration for high priority PUSH	S@T 1.20 V1.0.4	Accepted #29
2004-13	Axalto December 2003 #013	Session Id when commands follow a CONNECT_REQ	S@T 1.20 V2.0.0	Accepted (email vote 9 <sup>th</sup> March 2004)
2004-22	GEMPLUS-FEBRUARY-2004#5	Security level for administrative acknowledgement	S@T 1.20 V2.0.0	Accepted (email vote 9 <sup>th</sup> March 2004)
2004-27	G&D 02.2004 #5	12. Session diagram admin reference	S@T 1.20 V2.0.0	Accepted (email vote 9 <sup>th</sup> March 2004)
	PRISM-Dec-2007 #003	Clarifications for certification	S@T 1.20 V3.0.1	Accepted at WG meeting Jan 2008