

# Syrus and the Listener Server

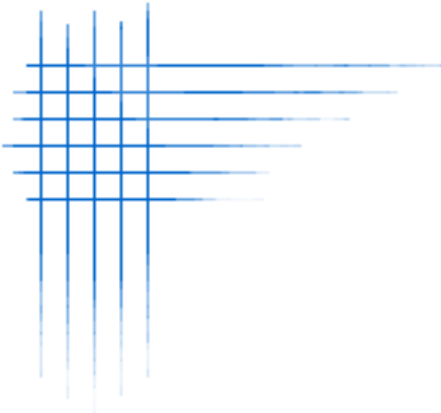
February 22<sup>th</sup> 2011

**DIGITAL COMMUNICATIONS TECHNOLOGIES**

**DIGITAL COMMUNICATIONS TECHNOLOGIES**

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





**DIGITAL COMMUNICATIONS TECHNOLOGIES**

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





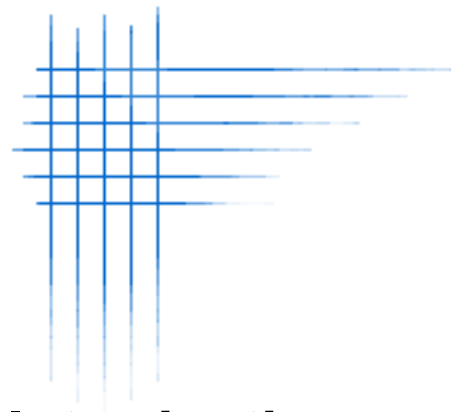
## Document history

Revision	Comments
001 - 23/12/2010	First release
002 – 22/02/2011	Added information about the AR configuration command.

## DIGITAL COMMUNICATIONS TECHNOLOGIES

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





## Introduction

This document is directed to Syrus-AVL-Sytem integrators designing the server program which receives and communicates with Syrus units via TCP and/or UDP.

This document lists important recommendations that must be followed to prevent data loss between the Syrus units and the listener server.

## DIGITAL COMMUNICATIONS TECHNOLOGIES

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>



## Types of messages sent by Syrus

Depending on the user set configuration, the Syrus units may report different types of messages. The listener server must be able to receive and correctly interpret the following types of messages.

- **Simple TAIP reports.** These are the regular TAIP messages sent by Syrus when an event is reported. These type of messages must always be parsed by the server. For example:

```
>REV111615353447+3570173+1397742703203012;ID=356612020083541<
```

Indicates that Syrus has reported event 11.

- **Extended TAIP reports.** These are the regular TAIP messages sent by Syrus when an event containing EV extended tags is reported. These type of messages and its extended tags must always be parsed by the server. For example:

```
>REV021615353839+3570173+1397742703203212;BL=4094;AL=+11;ID=Test<
```

Indicates that Syrus has reported event 02 which contains the BL and AL tags.

- **TAIP messages responses to queries.** These are messages sent by Syrus as a response to any query (i.e. >QPV<, >QAL<, >QVR<, etc) sent by the server. These messages may be parsed by the listener depending on the purpose of the query that was sent to the unit. For example:

The server sends a version query to the unit:

```
>QRV<
```

The unit will send the response:

```
>RVR; Syrus 1.1.39;ID=Test<
```

The listener shall process the message according to the purpose of the query

- **Unexpected TAIP responses to none-silent event-actions (i.e. events set with ;ACT=).** Syrus will send a TAIP message indicating that an action was executed when an event defined with non-silent actions is triggered. For example:

## DIGITAL COMMUNICATIONS TECHNOLOGIES

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulcation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>



The event definition:

```
>SED69XV0;C19+;ACT=SSSU080<
```

Will generate to the listener a normal event report:

```
>REV691615354941+3570173+1397742703203212;ID=Test<
```

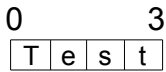
And will also generate a response for the executed action:

```
>RSSU080;ID=Test<
```

These messages may be parsed by the listener depending on the purpose of the executed action. Actions can be set as silent (i.e. ;XCT=) to prevent these types of messages. Please refer to the ED configuration command on the Syrus' manual.

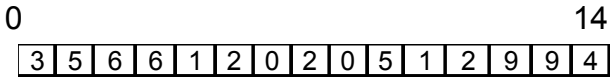
- **Keep alives.** The Syrus' Keep Alive message is sent after a period of communications inactivity. The keep alive message consist of a packet containing only the unit's ID (User defined ID string or the unit's IMEI) without the ">" or "<" characters. For example:

When the unit's ID is set as "Test" the keep alive message is sent as:



Total bytes received: 4

When the unit has its IMEI as ID:

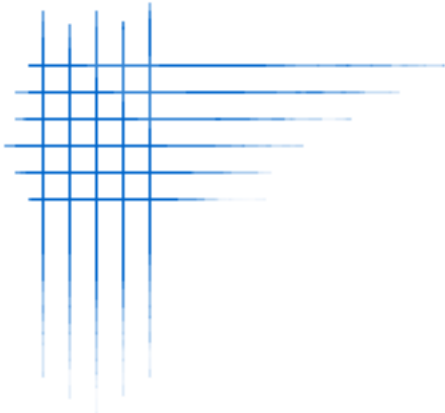


Total bytes received: 15

The listener server does not need to parse these messages, but the listener designer must be aware that they may arrive at any time and must silently discard these messages. Depending on the GPRS network conditions and the MTU (Maximum Transmission Unit) several messages may arrive in a single packet. For example:

### DIGITAL COMMUNICATIONS TECHNOLOGIES





0 29  

3	5	6	6	1	2	0	2	0	5	1	2	9	9	4	3	5	6	6	1	2	0	2	0	5	1	2	9	9	4
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Total bytes received: 30

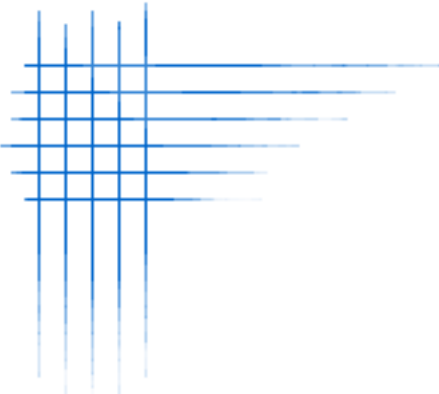
In this particular case they should also be ignored.

The Keep Alive inactivity time is set by the user configuration, and ranges between 1 and 15 minutes. When transmitted, it may contain the Carriage Return and Line Feed characters. The Carriage Return and Line Feed characters can be disabled using the XAKA Configuration Command. Please refer to the Syrus' manual for more information about this Configuration Command.

**DIGITAL COMMUNICATIONS TECHNOLOGIES**

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





**Extracting TAIP messages and reports from received TCP/UDP packets.**

Depending on GPRS network conditions, reporting criteria, and the **MTU** (Maximum Transmission Unit) of all networks connecting your server to Syrus devices, special cares must be taken to correctly receive all the data transmitted.

**Multiple TAIP messages may arrive in a single transmission packet.**

To extract the messages from the received transmission packets, it is best to parse the packet by extracting all data within ">" and "<" characters, ignoring everything else. It is important to realize that multiple TAIP messages can arrived in a single received packet. This means that multiple ">" and "<" character pairs **MUST BE** scanned. Otherwise some TAIP reports will silently get lost. For example:

Sometimes a single TAIP message may arrive in a single data packet:

0	31																																					
>	R	E	V	0	2	1	6	1	4	3	6	8	4	9	9	+	0	4	6	6	4	4	4	-	0	7	4	0	3	9	3							
8	0	0	0	0	0	1	1	;	V	O	=	0	;	B	L	=	4	0	5	9	;	D	O	P	=	-	,	-	,	-								
;	A	L	=	-	9	9	9	9	;	I	D	=	T	e	s	t	<	\r	\n																			

Total received bytes: 84

Sometimes two or more messages may arrive in a single packet, here shown in red and blue:

0	31																																				
>	R	X	A	I	T	2	0	0	0	E	V	0	0	0	0	0	0	0	0	0	0	0	0	+	0	0	0	0	0	0	0						
+	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	;	I	D	=	T	e	s	t	<	\r	\n	>	R	E	V						
0	2	1	6	1	4	3	6	8	5	0	1	+	0	4	6	6	4	4	4	-	0	7	4	0	3	9	3	8	0	0	0						
0	0	0	1	1	;	V	O	=	0	;	B	L	=	4	0	5	9	;	D	O	P	=	-	,	-	,	-	;	A	L	=						
-	9	9	9	9	;	I	D	=	T	e	s	t	<	\r	\n																						

Total received bytes: 144

Depending on the MTU, several messages could arrive in single data packet.

**DIGITAL COMMUNICATIONS TECHNOLOGIES**

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





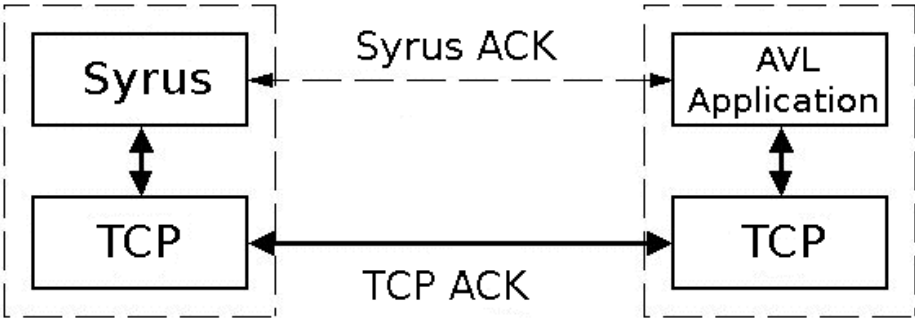


**Data confirmation mechanism**

It is recommended to always use the Syrus confirmation mechanism, both for TCP and UDP. TCP guarantees that the data was received but only at a connection level, it does not guarantee that the AVL application running in the server received the reports.

For example, when the data base of the application is temporarily unavailable but the listener is still working and receiving data from Syrus, the report will be confirmed as received at TCP level but the report could be lost because it was not saved in the data base. In this case, the listener should only confirm the report to Syrus when it was able to write it to the data base. The Syrus confirmation guarantees that the AVL application receives the reports.

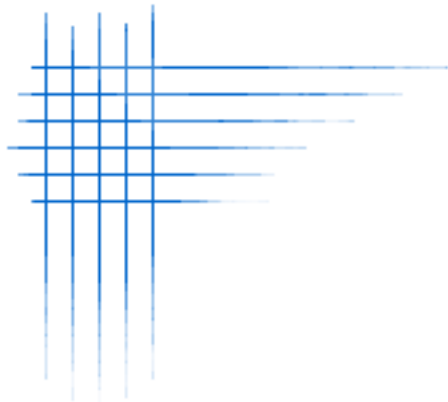
This mechanism will make Syrus keep sending a report message until it receives a confirmation from the listener server. The confirmation message sent by the server listener is the unit's ID without any other characters in the packet.



Syrus confirmation guarantees that the AVL Application received the reports.

**DIGITAL COMMUNICATIONS TECHNOLOGIES**





For example, When the unit sends a report like this:

0	31																														
>	R	E	V	0	2	1	6	1	4	3	6	8	4	9	9	+	0	4	6	6	4	4	4	-	0	7	4	0	3	9	3
8	0	0	0	0	0	0	1	1	;	V	O	=	0	;	B	L	=	4	0	5	9	;	D	O	P	=	-	,	-	,	-
;	A	L	=	-	9	9	9	9	;	I	D	=	T	e	s	t	<	\r	\n												

Total received bytes: 84

It will keep sending it until the server sends the confirmation which is the message:

0	3		
T	e	s	t

Total sent bytes: 4

If multiple events are received, they must all be confirmed.

**DIGITAL COMMUNICATIONS TECHNOLOGIES**

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulcation must be authorized.  
 December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>



## Syrus no-communications restart mechanisms

Syrus has several mechanisms that guarantees a two way communication with the listener server. One such mechanism is a Syrus System Reset after 8 hours (default time) of not receiving any data from any server set on a TCP/UDP Destination Point. This time can be changed using the AR configuration command. This configuration command also controls a System Reset when the unit has not sent any data to the server, also after a default time of 8 hours. These resets can be disabled or enabled independently,

The resets can be disabled completely, but it is recommended to use this mechanism because it ensures that the server is communicating with the unit and vice versa.

To prevent the resets while the mechanism is enabled (recommended) the AVL server must send any type of message to the unit, for example a position query, >QPV<. Any type of message before the 8 hours will prevent the unit from restarting. Although the reset is safely done by the unit (stored events data is saved), it is recommended to reduce them to a minimum, because during a System Reset, Syrus will pause the evaluation of events, hence some tracking and exceptional reports may not be triggered.

The AR command has the following format:

**ABBBBCDDDD**

Where:

**A** : Active flag:

0: Disable reception (RX) reset.

1: Enable reception (RX) reset.

**BBBB** : Time in minutes. Ranges from 30 to 1440 (24 hours). Default is 480 (8 hours).

**C** : Active flag:

0: Disable transmission (TX) reset.

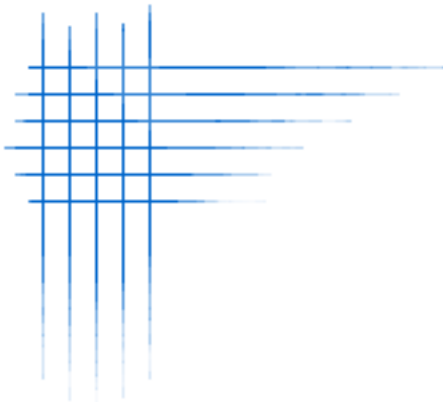
1: Enable transmission (TX) reset.

**DDDD** : Time in minutes. Ranges from 30 to 1440 (24 hours). Default is 480 (8 hours).

## DIGITAL COMMUNICATIONS TECHNOLOGIES

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulcation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





For example, to set a 10 hour reset when the unit does not receives data from the server and to disable the transmission reset, use this message:

```
>SRA1060000000<
```

To disable both resets, use the following message:

```
>SRA0000000000<
```

To enable both resets, using a wait time 5 hours when the unit does not receives data from the server, and a wait time of 2 hours when the unit does not send data to the server, use the following message:

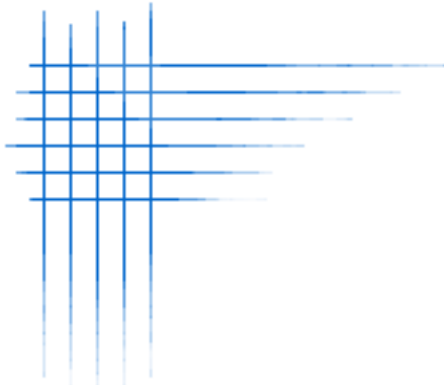
```
>SRA1030010120<
```

For more information about Syrus' Configuration Commands please refer to the Syrus User Manual.

## DIGITAL COMMUNICATIONS TECHNOLOGIES

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>





**END OF DOCUMENT**

**DIGITAL COMMUNICATIONS TECHNOLOGIES**

CONFIDENTIAL: This document is for the exclusive use of Digital Communications Technologies. All type of reproduction or divulgation must be authorized.  
December 23<sup>th</sup>, 2010 <http://www.digitalcomtech.com>

