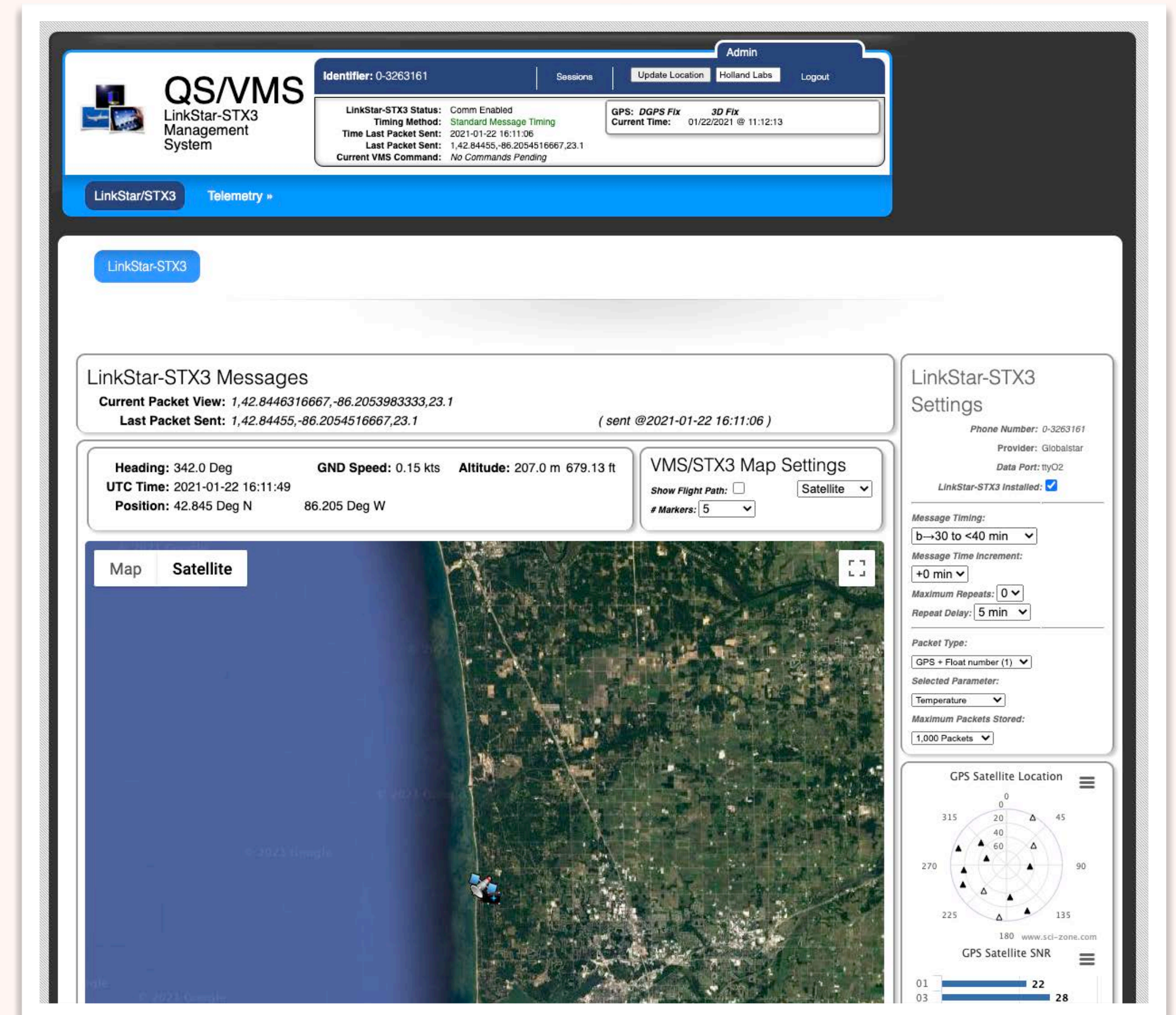

LinkStar-STX3 Radio System Series

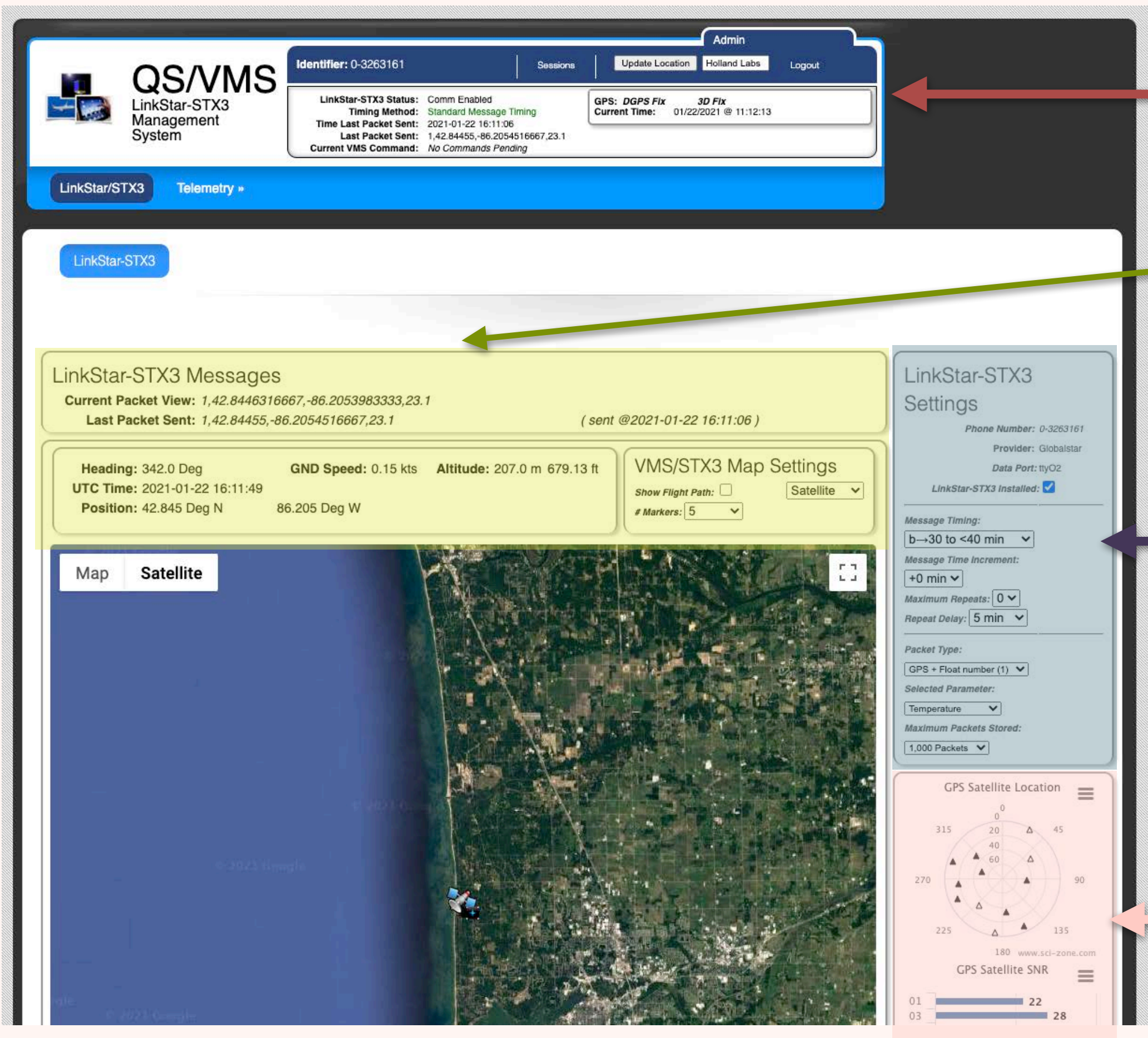
TRANSMITTING DATA

TRANSMITTING DATA - THE BASICS

- **Transmitting Data involves four key steps:**
 - **Adding custom parameters if required**
 - **Select your “Timing” of the messaging**
 - **Selecting the packet type to transmit**
 - **Select the Parameter**



THE LINKSTAR-STX3 HOME SCREEN



Common Header

Message and GPS Information

Radio Information and Radio Messaging

GPS Signal Quality Information

NOTE: LinkStar-STX3 Status MUST be “Comm Enabled” AND must have a GPS fix to transmit a message

Lower half of the screen presents historical data transmitted to the ground including the *Time Sent* (Universal Time), *Event Key* (unique message key on radio only), *Packet ID* (future feature), and the *Message* itself.

VMS/STX3 Message List

of Message Packets: 100 Erase Message List Search: Select all Deselect all

Time Sent	Event Key	Packet ID	Message
2021-01-22 16:11:06	25879	1	1,42.84455,-86.2054516667,23.1
2021-01-22 15:40:39	25878	1	1,42.844655,-86.2054783333,23.3
2021-01-22 15:10:13	25877	1	1,42.844785,-86.2056783333,22.8
2021-01-22 14:38:51	25876	1	1,42.8447716667,-86.20547,22.6
2021-01-22 14:07:02	25875	1	1,42.8446366667,-86.2054416667,22.9
2021-01-22 13:36:50	25874	1	1,42.8446566667,-86.2053416667,22.2
2021-01-22 13:06:24	25873	1	1,42.8444183333,-86.2051816667,22.0
2021-01-22 12:34:48	25872	1	1,42.84462,-86.2053916667,22.3
2021-01-22 12:04:20	25871	1	1,42.8446116667,-86.20534,22.2
2021-01-22 11:32:46	25870	1	1,42.8447266667,-86.2053066667,21.6
2021-01-22 11:01:50	25869	1	1,42.84468,-86.2053166667,21.2
2021-01-22 10:30:50	25868	1	1,42.8446633333,-86.20547,21.1
2021-01-22 09:58:53	25867	1	1,42.8446266667,-86.2054466667,21.1

Showing 1 to 100 of 100 entries

ADDING CUSTOM PARAMETERS...

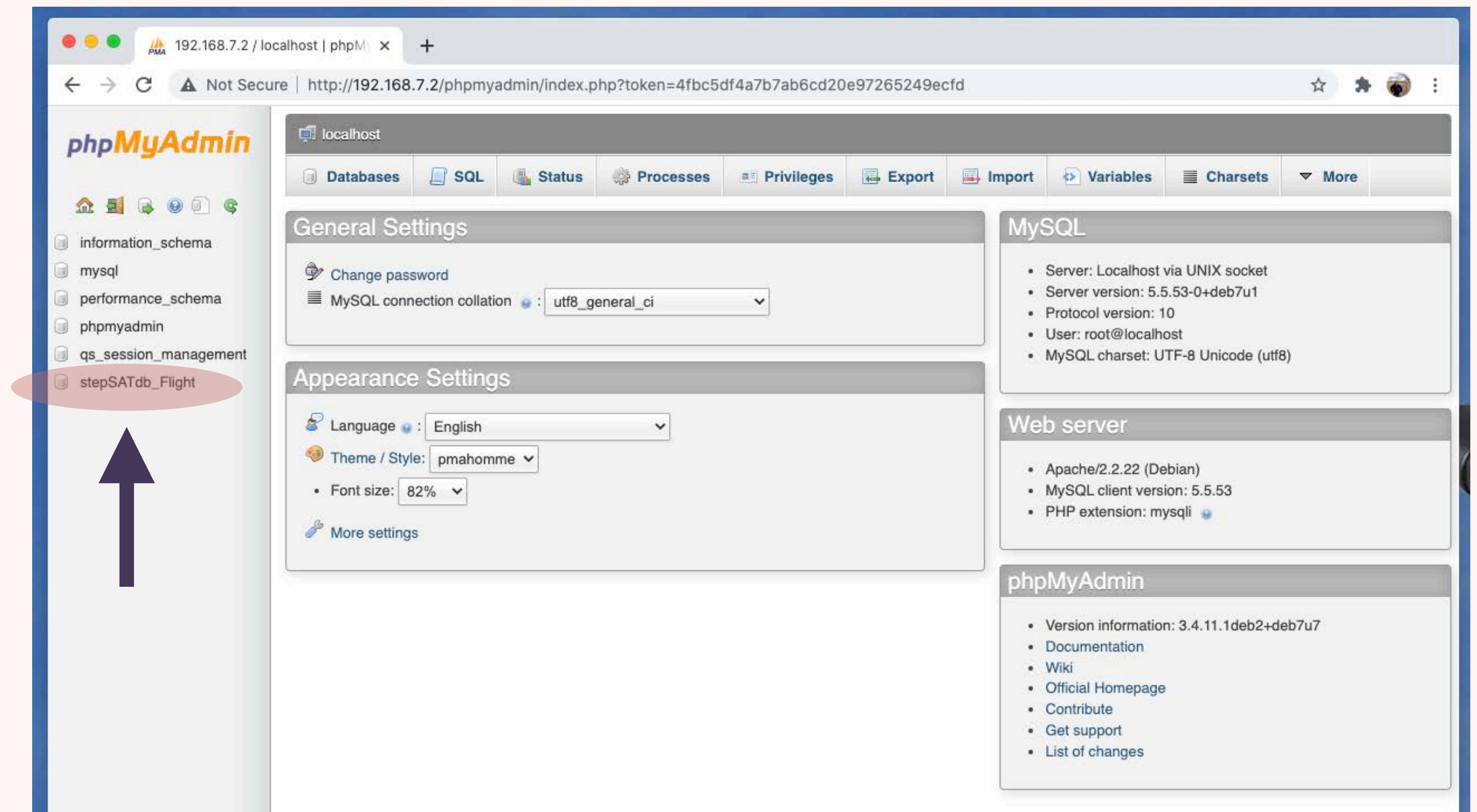
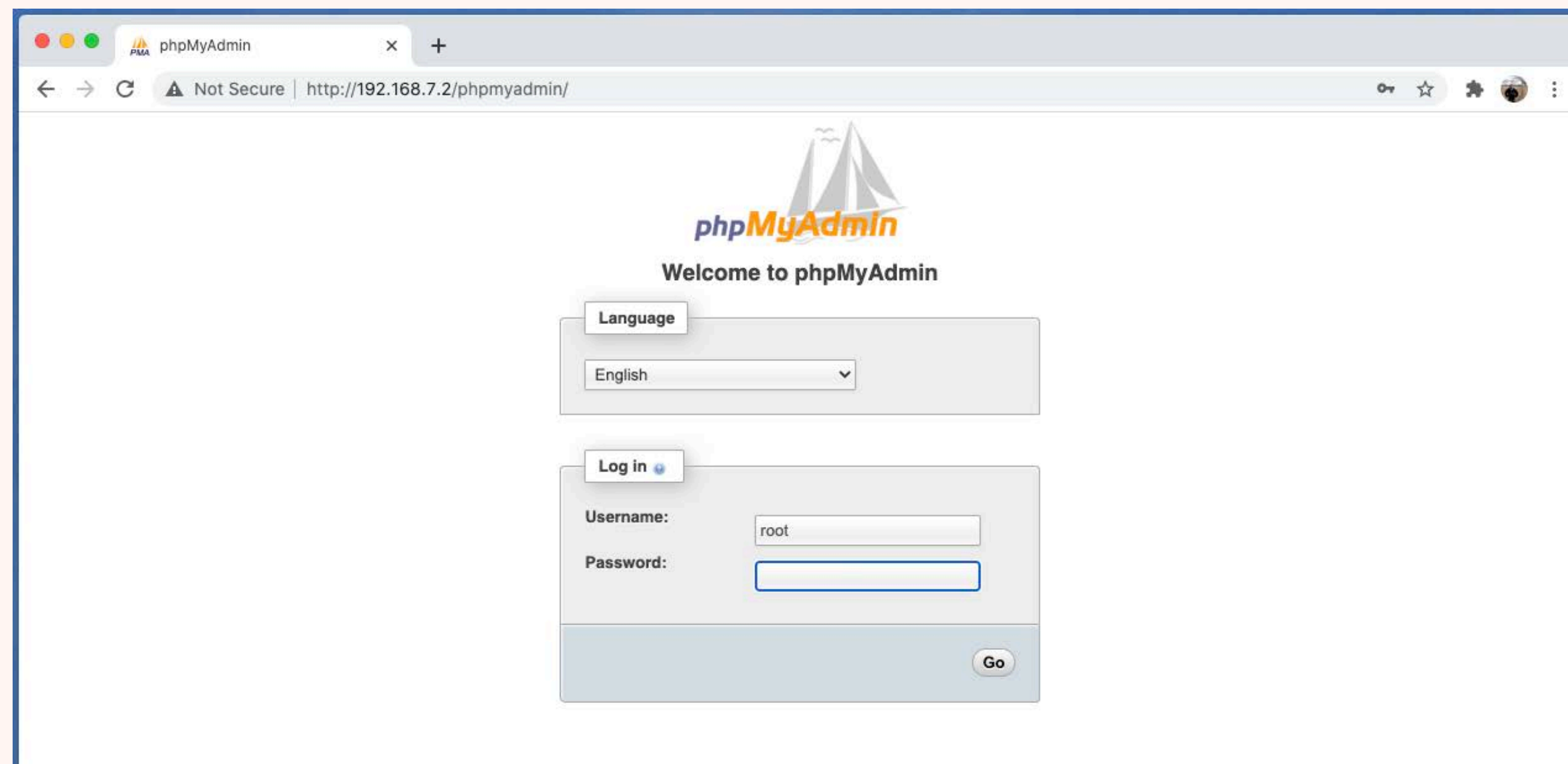
➤ Current Steps:

➤ Using web browser go to <http://192.168.7.2/phpmyadmin>

➤ Login -> Username: root Password: Quicksat!1

➤ Select *step_SATdb_Flight*

Yes, you CAN CHANGE the username and password for PHPMYADMIN and MYSQL. The user can find how to change these values in the supporting PHPMYADMIN, MySQL and Debian Linux documentation.



ADDING CUSTOM PARAMETERS...

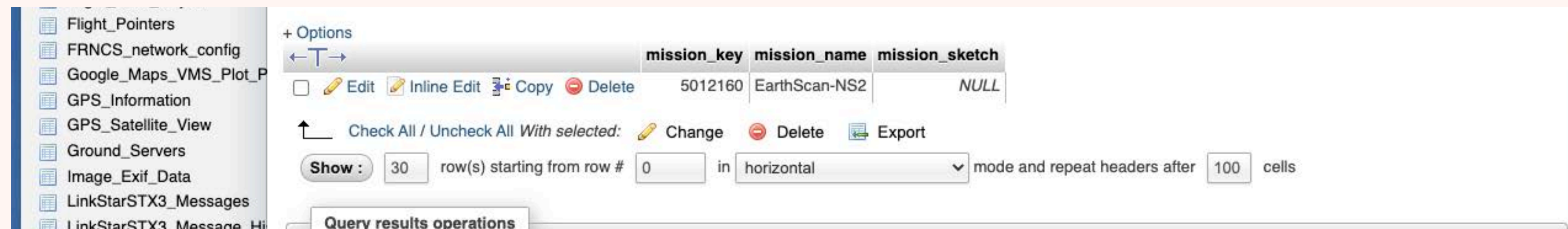
- Using web browser go to <http://192.168.7.2/phpmyadmin>
- Login -> Username: root Password: Quicksat!1
- Select *step_SATdb_Flight*



This screenshot shows the 'step_SATdb_Flight' database selected in the left sidebar. The main area displays a list of tables. The 'Mission' table is highlighted, and an arrow points to it from the text 'Select the "Mission" Table'.

Table	Engine	Character Set	Collation	Row Count	Size
LinkStarSMS	InnoDB	latin1_swedish_ci		1,001	112.0 KiB
LinkStarSTX3_Message_History	InnoDB	latin1_swedish_ci		1	2.0 KiB
LinkStar_Duplex_Information	InnoDB	latin1_swedish_ci		0	16.0 KiB
LinkStar_Duplex_State	InnoDB	latin1_swedish_ci		0	16.0 KiB

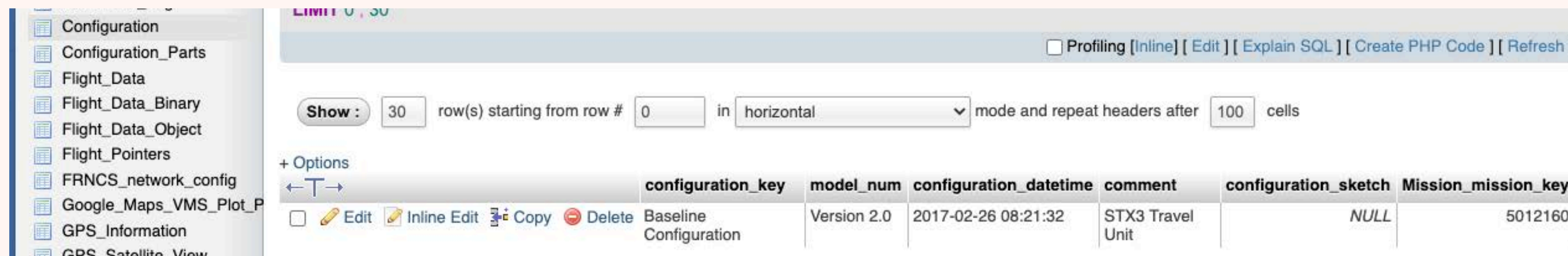
Select the "Mission" Table



This screenshot shows the 'Mission' table selected. The table structure and data are visible. The 'mission_key' column is highlighted, and an arrow points to it from the text 'Note the "mission_key". You will use this to define your parameters. You change this value AND change the "mission_name" using Edit or Inline Edit, but use the same values consistently that reference these columns.'

mission_key	mission_name	mission_sketch
5012160	EarthScan-NS2	NULL

Note the "mission_key". You will use this to define your parameters. You change this value AND change the "mission_name" using *Edit* or *Inline Edit*, but use the same values consistently that reference these columns.



This screenshot shows the 'Configuration' table selected. The table structure and data are visible. The 'Mission_mission_key' column is highlighted, and an arrow points to it from the text 'Next select the "Configuration" Table. Note the "configuration_key" and verify "Mission_mission_key" matches the "mission_key" in the Mission table. If not, Edit or Inline Edit as needed.'

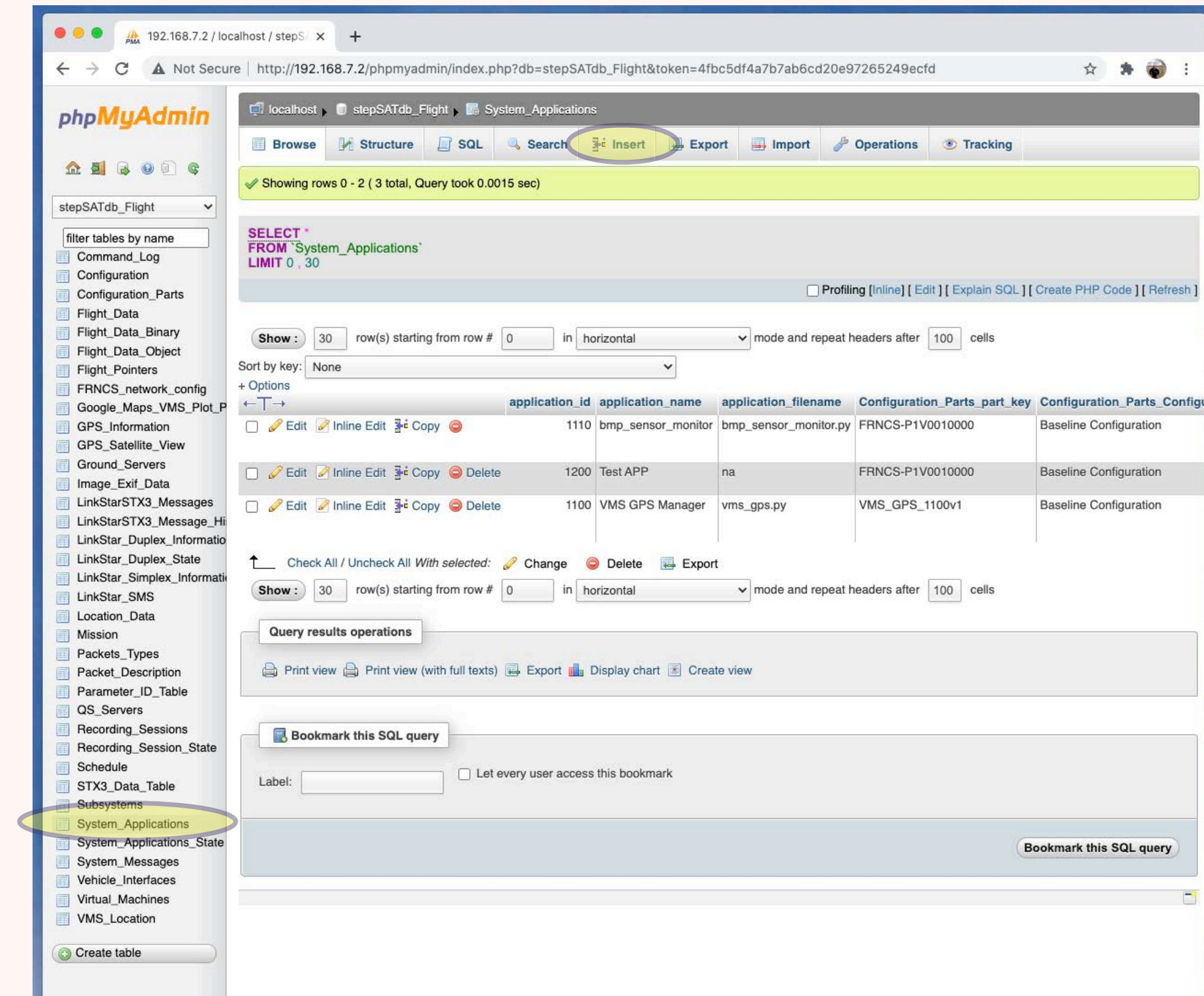
configuration_key	model_num	configuration_datetime	comment	configuration_sketch	Mission_mission_key
Baseline Configuration	Version 2.0	2017-02-26 08:21:32	STX3 Travel Unit	NULL	5012160

Next select the "Configuration" Table.

Note the "configuration_key" and verify "Mission_mission_key" matches the "mission_key" in the Mission table. If not, *Edit* or *Inline Edit* as needed.

ADDING CUSTOM PARAMETERS...

- **Next step is to define the *software application* containing the customer, user defined parameters**
- **The *software application* is referenced to your defined “Configuration”...referenced to the defined “Mission”**
- ***Software applications* have a set of Parameters**
- **Data generated from the *software application* is referenced to a *Parameter***
- **Select the “System_Applications” Table**
- **Select “Insert” near the top**



ADDING CUSTOM PARAMETERS...

➤ *Insert the Software Application in* **“System_Applications”**

Insert a custom “application_id” that you define

Give your application a name

Enter the filename of the application
Software applications are to be stored in the directory /opt/qs/bin

Assign a unique “Configuration_Parts_part_key” database key
for you application (optional)

Enter the “configuration_key” in the
“Configurations_Parts_Configuration_configuration_key”

Enter the “mission_key” in the
“Configuration_Parts_Configuration_Mission_mission_key”

Enter the “description” of the software application

Enter -1 for the “virtual_machine_id”

Enter 1 for the “has_parameters”...you telling QuickSAT/VMS your
software application has parameters!

The screenshot shows the phpMyAdmin interface for the 'stepSATdb_Flight' database, specifically the 'System_Applications' table. The table structure is displayed with columns and their corresponding data types and values. Annotations with arrows point from text boxes to specific fields in the table:

Column	Type	Function	Null	Value
application_id	int(11)			1500
application_name	varchar(45)			My Great Application
application_filename	varchar(100)			mygreatapp.py
Configuration_Parts_part_key	varchar(50)			mygr_v00100
Configuration_Parts_Configuration_configuration_key	varchar(40)			Baseline Configuration
Configuration_Parts_Configuration_Mission_mission_key	int(11)			5012160
application_state	int(11)			100
application_status	varchar(45)			Operational - Monitoring
description	varchar(45)			My Great Test Application doing its great thing
virtual_machine_id	int(11)			-1
has_parameters	tinyint(4)			1
locked_flag	tinyint(1)			1
software_part_number	varchar(45)			mygr_v00100
target_board_installed	tinyint(1)			1
on_off_switch	tinyint(1)			0

ADDING CUSTOM PARAMETERS...

Define the “application state” and “application_status” - NOTE: your software_application CAN change this value to allow constant monitoring of the status of your software application

application_state	application_status
50	Ground Storage
80	FRNC Storage
100	Operational
110	Paused
190	VM Initializing
195	Installed On Host
150-189, 191-194, 196-199	Reserved
200	Off
205	Unable To Start Application
250-299	Reserved
300	Error
305	No Contact With Application
350-399	Reserved
400-699	<i>Open to users to define</i>
700-999	Reserved

Future Feature - set the software application to locked

Enter the “software_part_number” of the software application (optional)

Enter 1 to tell QuickSAT/VMS your software is installed

Future Feature - you can enable an “on/off” switch to start and stop your software application.

The screenshot shows the phpMyAdmin interface for the 'stepSATdb_Flight' database. The 'System_Applications' table structure is displayed. The columns and their values are as follows:

Column	Type	Function	Null	Value
application_id	int(11)			1500
application_name	varchar(45)			My Great Application
application_filename	varchar(100)			mygreatapp.py
Configuration_Parts_part_key	varchar(50)			mygr_v00100
Configuration_Parts_Configuration_configuration_key	varchar(40)			Baseline Configuration
Configuration_Parts_Configuration_Mission_mission_key	int(11)			5012160
application_state	int(11)			100
application_status	varchar(45)			Operational - Monitoring
description	varchar(45)			My Great Test Application doing its great thing
virtual_machine_id	int(11)			-1
has_parameters	tinyint(4)			1
locked_flag	tinyint(1)			1
software_part_number	varchar(45)			mygr_v00100
target_board_installed	tinyint(1)			1
on_off_switch	tinyint(1)			0

ADDING CUSTOM PARAMETERS...

- Once the *software application* has been added to “System Applications” you can now add the parameters for the *system application* to the table “Parameter_ID_Table”
- Select the table “Parameter_ID_Table”
- Select *Insert* insert the *parameters*

The screenshot shows the phpMyAdmin interface for the 'stepSATdb_Flight' database. The 'Parameter_ID_Table' is selected, and the 'Insert' button is highlighted. The table structure is displayed below the SQL query area.

parameter_id	System_Applications_application_id	parameter_name	units	type	description	conversion	stx3_selected_parameter	table_location
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1110	Temperature	C	FLOAT	Temperature Sensor data from BMP Sensor	n/a	1	Data
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1110	Pressure	Pa	FLOAT	Pressure Sensor data from BMP Sensor	n/a	0	Data
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1110	Altitude	meters	FLOAT	Calculated based on Pressure Sensor data from BMP ...	n/a	0	Data
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1110	Sea Level Pressure	Pa	FLOAT	Sea Level Pressure Sensor data from BMP Sensor	n/a	0	Data
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1200	1000-B		TEXT	Block data from demo test application	n/a	0	Data_Object
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1100	Speed-GPS	kts	FLOAT	Speed of the LinkStar-STX3	n/a	0	Data
<input type="checkbox"/> Edit <input type="checkbox"/> Inline Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1100	Altitude-GPS	meters	FLOAT	Altitude of the LinkStar-STX3	n/a	0	Data

ADDING CUSTOM PARAMETERS...

➤ **Insert the information for the parameter in the “Parameter_ID_Table”**

Insert a unique “parameter_id” that you define

Enter the “application_id” of the parameter’s *software application*

Enter the name of the parameter

Enter the units of the parameter

Enter the “type” of the parameter. Types include INTEGER, FLOAT, TEXT; BINARY, TIF, JPG, PNG, RAW can ONLY be used for LinkStar duplex radios.

Enter the “description” of the parameter (optional)

Enter the conversion factor of the parameter (optional). This is a text, pseudo code of the method of conversion your *software application* can use

Enter 0 for the “stx3_selected_parameter”

Select which table, or “table_location”, where you are writing the value/data of the parameter too. Your options are “Data” for the “Flight_Data” table, “Data_Object” for “Flight_Data_Object”; “Data_Binary” for “Flight_Data_Binary” is only available for LinkStar duplex radios

The screenshot shows the phpMyAdmin interface for the 'stepSATdb_Flight' database. The 'Parameter_ID_Table' structure is displayed, showing columns: parameter_id (varchar(45)), System_Applications_application_id (int(11)), parameter_name (varchar(45)), units (varchar(10)), type (varchar(15)), description (varchar(80)), conversion (varchar(150)), stx3_selected_parameter (tinyint(1)), and table_location (enum). Below the structure, the 'Insert' form is shown with fields for each column. Arrows point from the text boxes to the corresponding fields in the form.

Column	Type	Function	Null	Value
parameter_id	varchar(45)			
System_Applications_application_id	int(11)			
parameter_name	varchar(45)		<input checked="" type="checkbox"/>	
units	varchar(10)		<input checked="" type="checkbox"/>	
type	varchar(15)		<input checked="" type="checkbox"/>	
description	varchar(80)		<input checked="" type="checkbox"/>	
conversion	varchar(150)		<input checked="" type="checkbox"/>	
stx3_selected_parameter	tinyint(1)		<input type="checkbox"/>	0
table_location	enum	--	<input type="checkbox"/>	Data

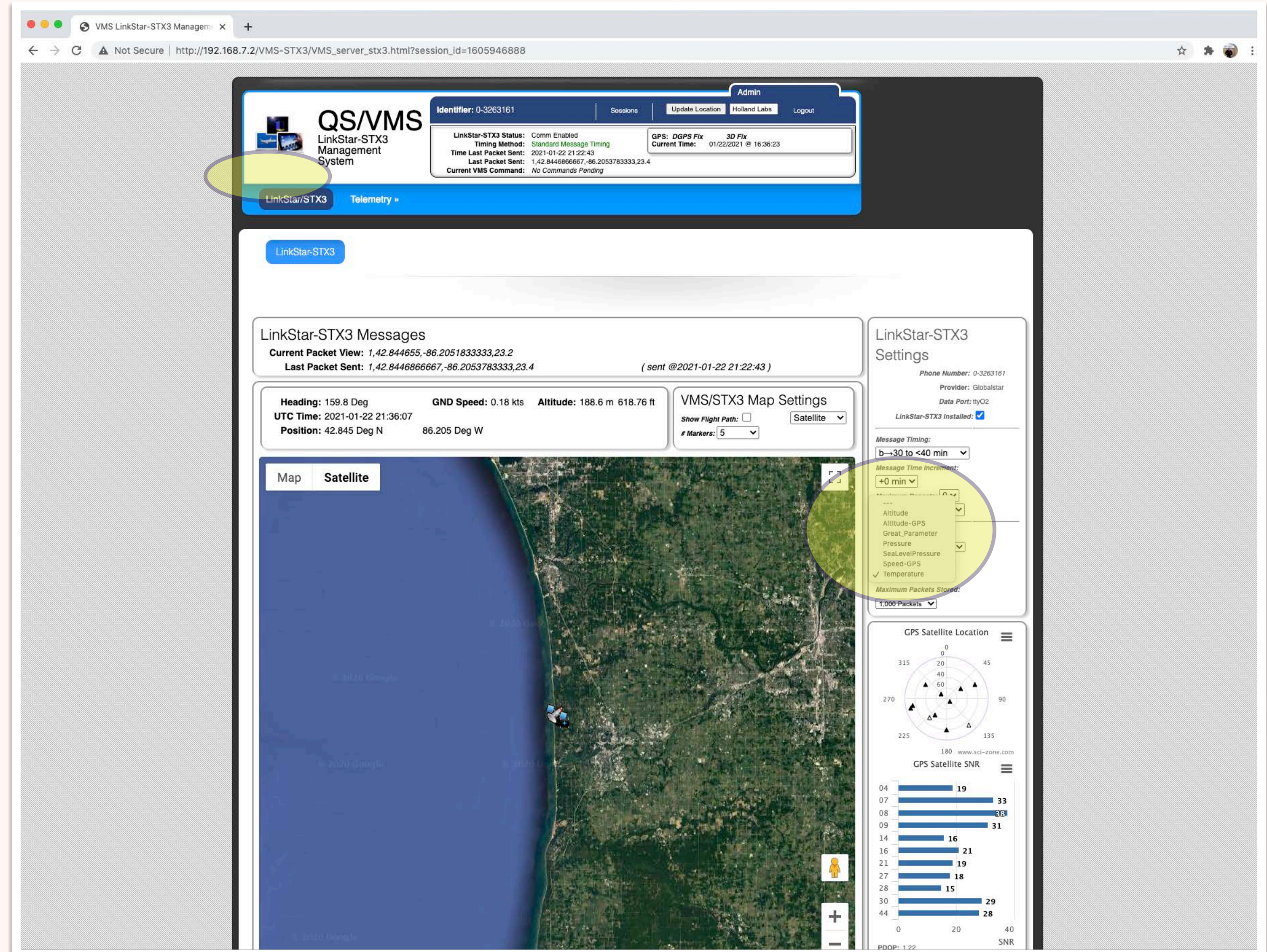
Go

☒ Ignore

Column	Type	Function	Null	Value
parameter_id	varchar(45)			
System_Applications_application_id	int(11)			
parameter_name	varchar(45)		<input checked="" type="checkbox"/>	

ADDING CUSTOM PARAMETERS...

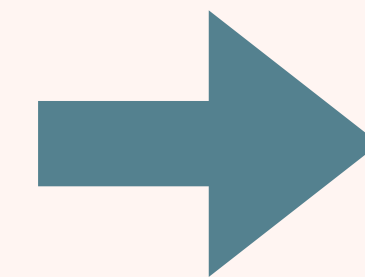
- Verify your parameter is being tracked by the *QuickSAT/VMS* system
- Go to the LinkStarSTX3 tab (the “Home Screen”)
- Under *LinkStar-STX3 Settings* select the *Packet Type* of “GPS+Integer” to view parameters with a “table_location” of *Data*, or “GPS+Text Message” for parameters with “table_location” of *Data_Object*.
- Select the drop down menu under “Selected Parameter” - your new parameter should now be listed there!



SELECT THE “TIMING” OF YOUR MESSAGE

- There are several options on how often you can beacon your message, and how many times it can be repeated. Messaging patterns are per FCC, ITU and Globalstar regulations.

Message Type	Message Timing	Message Time Increment	Maximum Repeats	Repeat Delay
a	40 to 1400 min	+0,1,2,3,4,5,6,7,8,9 minutes	2	5, 6, 7, 8, 9, 10 minutes
b	30 to <40 min	+0,1,2,3,4,5,6,7,8,9 minutes	2	5, 6, 7, 8, 9, 10 minutes
c	20 to <30 min	+0,1,2,3,4,5,6,7,8,9 minutes	1	5, 6, 7, 8, 9, 10 minutes
d	10 to <20 min	+0,1,2,3,4,5,6,7,8,9 minutes	NONE	NONE
e	ALARM MODE (5 min)	+0,1,2,3,4 minutes	NONE	NONE
f	Balloon MODE (2 min) for 12 hours and then the message is permanently stopped	+0,1,2,3,4 minutes	NONE	NONE



LinkStar-STX3
Settings

Phone Number: 0-3263161
Provider: Globalstar
Data Port: ttyO2
LinkStar-STX3 Installed: ☒

Message Timing:
b→30 to <40 min

Message Time Increment:
+0 min

Maximum Repeats: 0

Repeat Delay: 5 min

Packet Type:
GPS + Text Message (\$)

Selected Parameter:

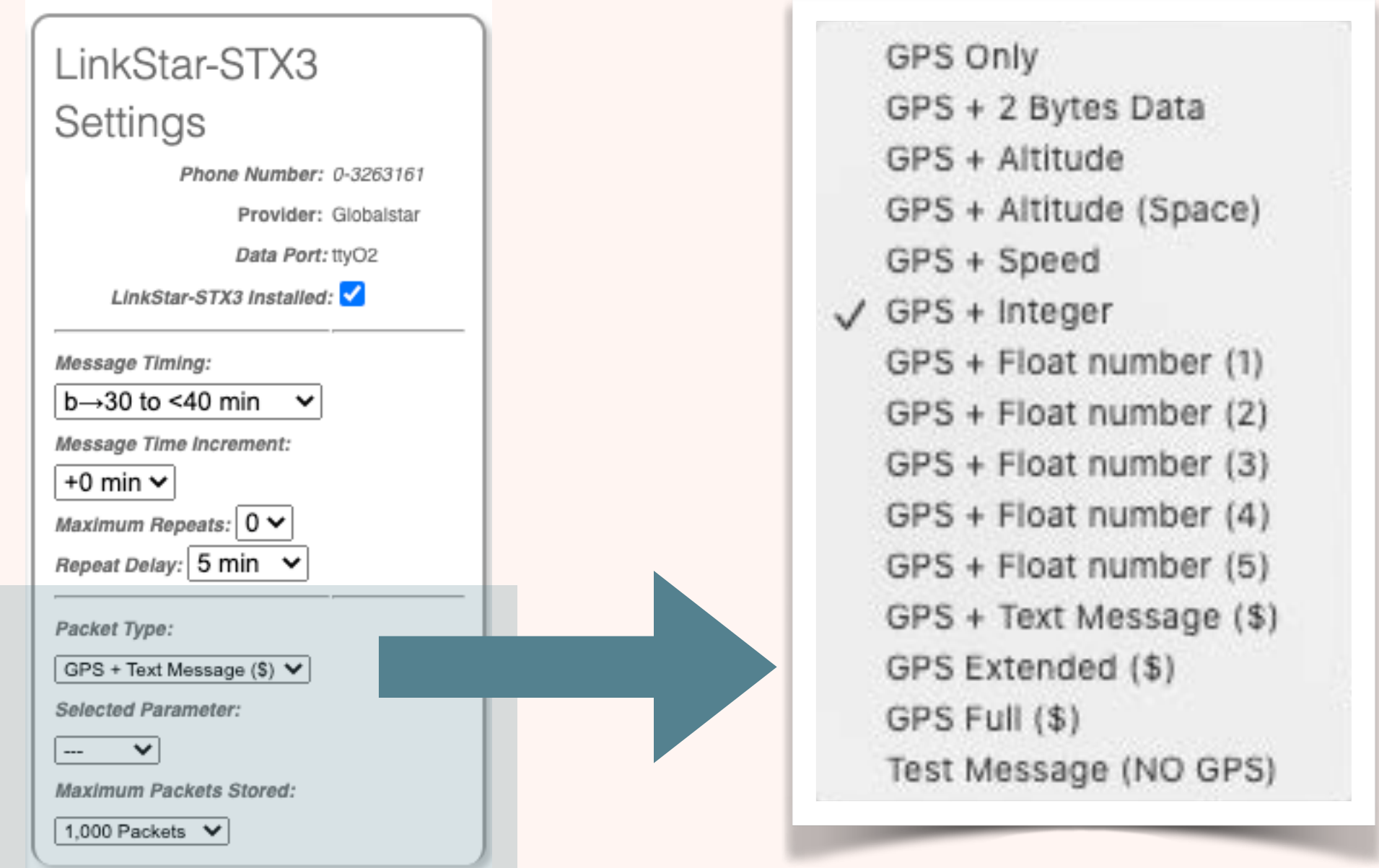
Maximum Packets Stored:
1,000 Packets

Examples:

- ✳ Selecting *Message Type (a)* with a *Message Time Increment* of 4 minutes and *Maximum Repeats* of 2 with a *Repeat Delay* of 5 minutes your message will be broadcasted approximately at **44 minutes** and repeated at **49 minutes** and **54 minutes** from the prior broadcast.
- ✳ Selecting *Message Type (c)* with a *Message Time Increment* of 5 minutes and *Maximum Repeats* of 1 with a *Repeat Delay* of 10 minutes your message will be broadcasted approximately at **25 minutes** and repeated at **35 minutes** from the prior broadcast.
- ✳ Selecting *Message Type (d)* with a *Message Time Increment* of 2 minutes your message will be broadcasted approximately at **12 minutes** from the prior broadcast.
- ✳ Selecting *Message Type (e)* with a *Message Time Increment* of 0 minutes your message will be broadcasted approximately at **5 minutes** from the prior broadcast.
- ✳ Selecting *Message Type (f)* with a *Message Time Increment* of 0 minutes your message will be broadcasted approximately at **2 minutes** from the prior broadcast. The messaging will stop permanently 12 hours from the first broadcast.

SELECT THE “PACKET TYPE” OF YOUR MESSAGE

➤ There are several pre-set packet types to select from. If you think you will be able to recover your LinkStar simplex radio select to broadcast the data you feel is critical to broadcast. The rest of the data can be viewed later connecting your radio to your desktop. Otherwise, broadcast all data you feel is critical to view remotely



Packet Type	Description
GPS Only	Only Latitude and Longitude
GPS + 2 Bytes Data	Latitude and Longitude plus 2 Bytes of data in character format - can be converted to individual bits to extract
GPS + Altitude	Latitude and Longitude plus Altitude in m up to 65535 m unsigned
GPS + Altitude (Space)	Latitude and Longitude plus Altitude in km up to 1500 km unsigned
GPS + Speed	Latitude and Longitude plus Speed in kts up to 65535 kts unsigned
GPS + Integer	Latitude and Longitude plus an Integer value from the Selected Parameter, Data , to ± 65535
GPS + Float Number (1)	Latitude and Longitude plus an Integer value from the Selected Parameter, Data , to ± 6553.5
GPS + Float Number (2)	Latitude and Longitude plus an Integer value from the Selected Parameter, Data , to ± 655.35
GPS + Float Number (3)	Latitude and Longitude plus an Integer value from the Selected Parameter, Data , to ± 65.535
GPS + Float Number (4)	Latitude and Longitude plus an Integer value from the Selected Parameter, Data , to ± 6.5535
GPS + Float Number (5)	Latitude and Longitude plus an Integer value from the Selected Parameter, Data , to ± 0.65535
GPS + Text Message (\$)	Latitude and Longitude plus an Integer value from the Selected Parameter, Data Object , up to 137 bytes
GPS + Extended Message (\$)	UT Time Stamp, Latitude, Longitude, Speed (kts), Altitude (m)
GPS + Full Message (\$)	UT Time Stamp, Latitude, Longitude, Speed (kts), Altitude (m), Heading, Fix Type, Fix Value
Test Message (NO GPS)	UT Time Stamp

SELECT THE PARAMETER

- Once your *Timing* is set along with the *Packet Type* and you want to broadcast **Byte, Integer, or Float Number** data you need to select the *Parameter*.

LinkStar-STX3
Settings

Phone Number: 0-3263161
Provider: Globalstar
Data Port: ttyO2
LinkStar-STX3 Installed: ☒

Message Timing:
d→10 to <20 min ▼

Message Time Increment:
+0 min ▼
- No Repeat Messages Allowed

Packet Type:
GPS + Integer ▼

Selected Parameter:
Temperature ▼

Maximum Packets Stored:
1,000 Packets ▼

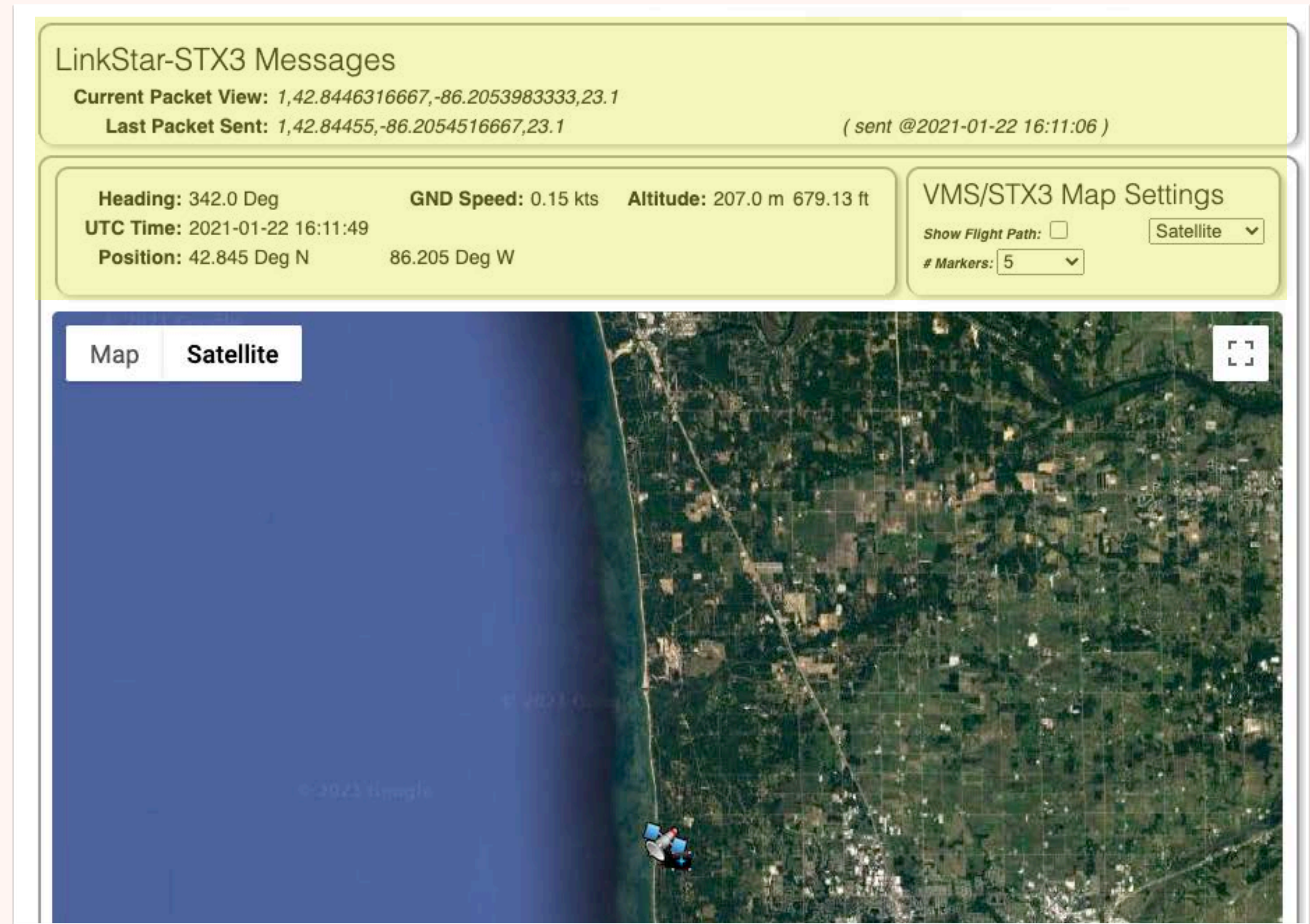
Go to the *Selected Parameter* drop down menu

You can also select the number of prior packets transmitted to be stored on the onboard computer. Messages are deleted "First In/Last Out"

Altitude
Altitude-GPS
Pressure
SeaLevelPressure
Speed-GPS
✓ Temperature

VIEW MESSAGE FORMAT AND STATUS

- To verify your message format look at the “LinkSar-STX3 Messages” area
- “Current Packet View” shows how the packet will look at that moment (updated every 30 seconds) based on your “Packet Type” and if selected, your Parameter”
- “Last Packet Sent” shows the exact format of the packet transmitted to the ground.
- GPS Data is also presented underneath the LinkStar-STX3 Messages area.



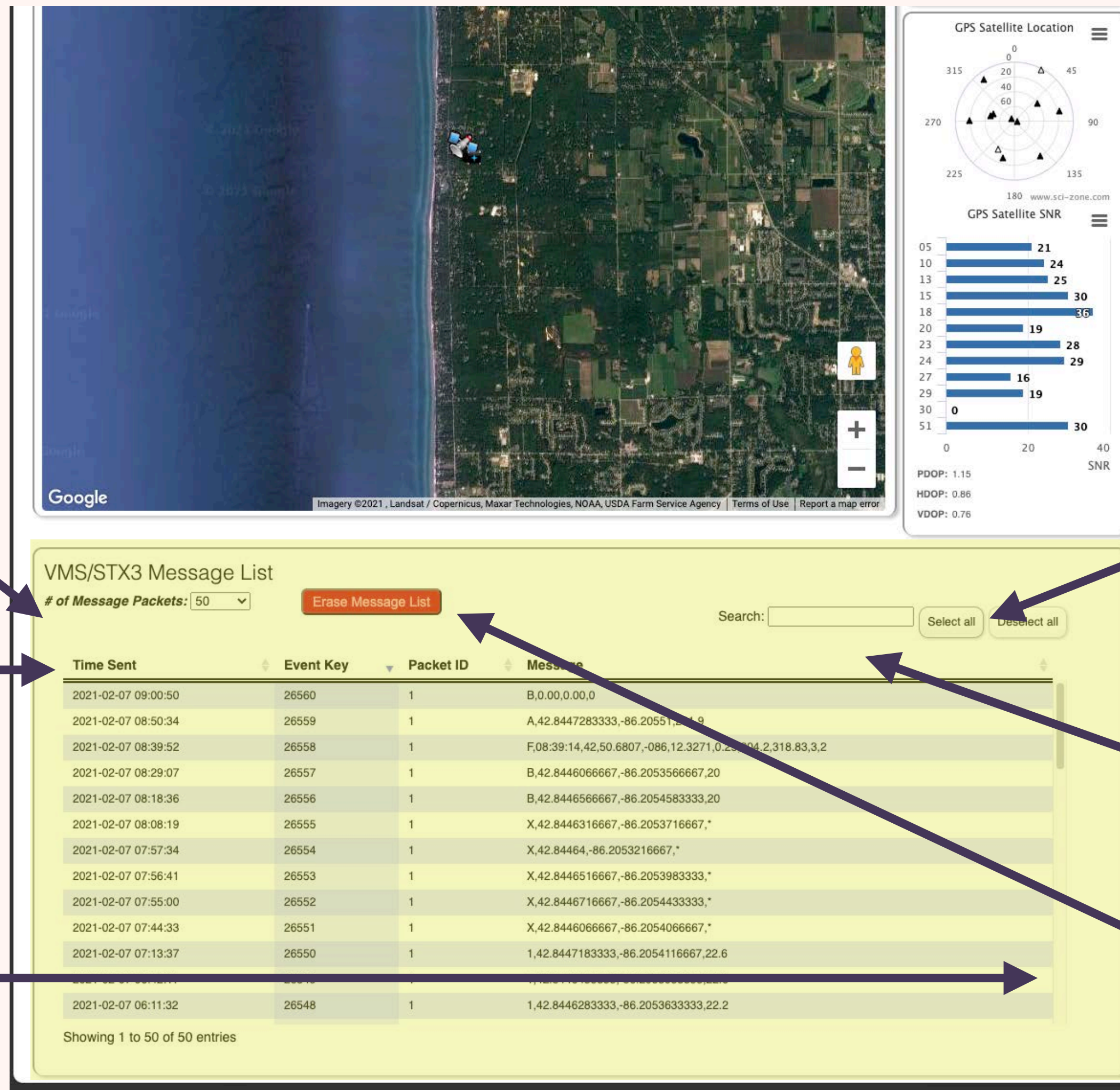
VIEW THE DATA - MESSAGE HISTORY

➤ At the bottom half of the **QuickSAT/VMS** “home screen” (LinkStar/STX3) you can view the history of messages transmitted under “**VMS/STX3 Message List**”

You can select the number of packets to view

You can click on the column headings to change the order of the data viewed in ascending or descending order

You can scroll through the table with your mouse.



You can click on a row to select, “Select All”, “Deselect All” rows for copying or export to CSV, Excel or PDF (buttons for the export functions are coming out in the next release)

You can search for a value throughout the entire message list

For security purposes you can erase the entire message list