1. GRASS-RaPlaT for experiment planning and visualization of measurements

In the LOG-a-TEC testbed, GRASS-RaPlaT is used to enhance the testbed with (i) additional GIS functionalities, (ii) transmitter radio coverage calculation/visualization and (iii) support for radio environment map estimation. In order to achieve the above mentioned functionalities, the geographical area of the LOG-a-TEC testbed is added to the GRASS-RaPlaT tool as a standalone GRASS mapset, with all the necessary layers for pathloss calculation including the digital elevation model (DEM), clutter file containing the terrain usage, etc.

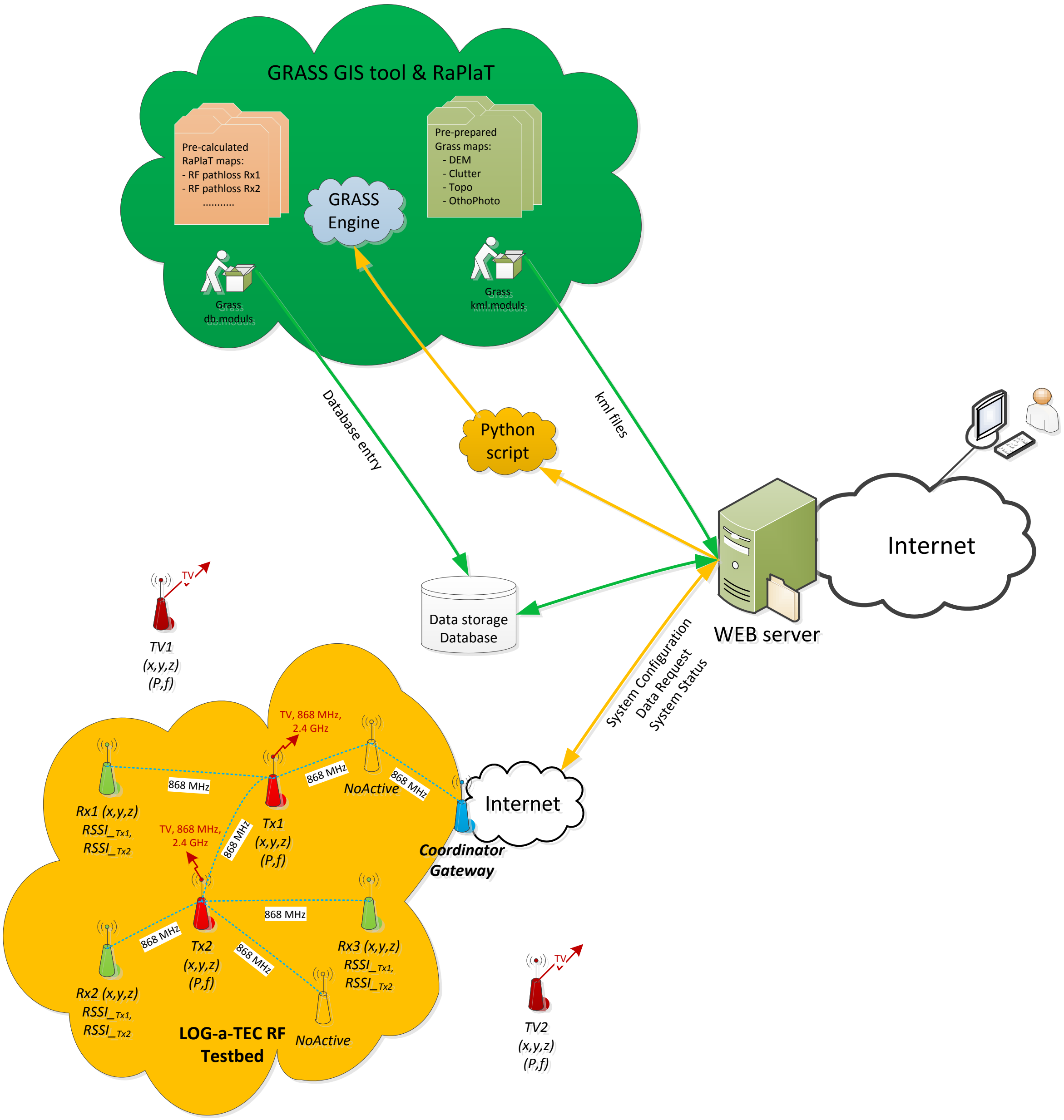


Figure 1 Integration of the GRASS-RaPlaT in LOG-a-TEC testbed

The integration of GRASS RaPlaT in the LOG-a-TEC testbed is depicted in Figure 1. The testbed consists of the following functional blocks (i) LOG-a-TEC RF testbed, GRASS GIS with RaPlaT, (ii) data storage/database and (iii) WEB server.

Two initial functionalities implemented within GRASS RaPlaT to support experiments are:

(a) Tx range estimation by specifying Rx sensitivity and Tx transmit power and
(b) estimation of the received signal strength at the specified Rx taking into consideration all active transmitters.

These two basic functionalities will help the experimenter to select the number of transmitters, their transmit powers and locations, as well as the locations of the receivers, which best suit the requirements of the experiment. The basic functionality (b) can be upgraded during the experiment to show also the RSSI measurement results retrieved from the Rx nodes. In order to find whitespaces, the basic functionality (a) can be upgraded to show the coverage areas of all active transmitters.

The basic functionalities are shown in Figure 2.

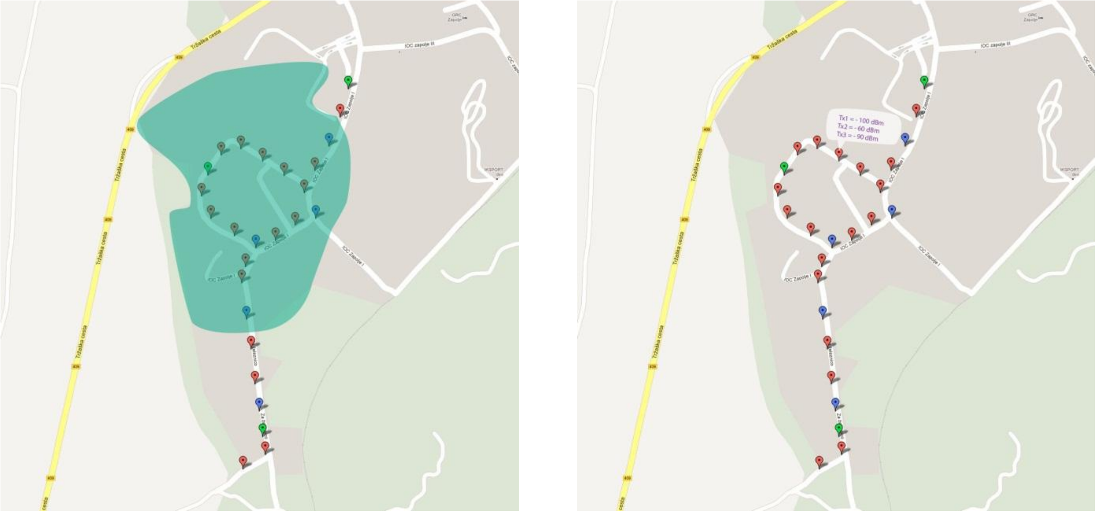


Figure 2 The basic GRASS RaPlaT functionalities for supporting the experiment setup

1.1 LOG-a-TEC WEB portal - GRASS-RaPlaT intercommunication

(Information given here supersedes preliminary specifications for *wsn.coverage* given in Project Deliverable D3.3.)

The LOG-a-TEC web portal requests GRASS-RaPlaT computations by issuing specially prepared Linux commands. The portal first prepares the necessary file(s) with input data for GRASS-RaPlaT and then calls the corresponding Linux command with additional data given as command line parameters. The command returns the computed data in a file. It returns a zero value in case of success or a non-zero value otherwise. It usually writes additional information about the execution on stdout and stderr, but this can generally be ignored by the portal (except for debugging).

Currently, GRASS-RaPLAT supports the following computations for the portal:

1. Coverage calculations for one or more transmission points (sensor nodes). It returns a KMZ file containing a raster picture with color-coded representation of the strongest received signal for each raster point. Included is also an additional raster picture containing the legend. The calculation for each transmission point is limited by the radius given as a command line parameter (points outside the radius remain transparent).
2. The same as above, but showing only the points with the received power above a given threshold value (the rest are transparent).
3. The same as above, but the covered points (those within calculation radius and above the optional threshold value) are coloured with a single colour (currently red).
4. The calculation of the received powers from all transmitters for a list of receive locations specified in a file. The result is a file with a list of received powers from all transmitters for each given receive location.

The described computations are performed by two Linux commands, *wsn.coverage* (compuatations 1 to 3) and *wsn.rxpowers* (computation 4). These commands are used in the normal Linux environment outside the GRASS environment. They transiently establish GRASS environment needed for executing the necessary GRASS-RaPlaT commands.

1.1.1 Coverage computation (command *wsn.coverage*)

Web server requests coverage computation by using the following Linux command:

```
wsn.coverage --input=<input_file> --output=<output_file> --radius=<radius_in_km> \
[--rx_threshold=<rx_threshold_in_dBm>] [-1] [--help]
```

or in short form:

```
wsn.coverage -i <input_file> -o <output_file> -r <radius_in_km> [-t=<rx_threshold_in_dBm>] [-1] [-h]
```

The command returns 0 upon successful completion or an error code otherwise. It may also print some descriptive information on stdout/stderr, which the Web portal can safely ignore.

Usage (examples)

The following input file *nodes-868-short* is used in the examples:

```
e           ; n           ; h   ; tx_frequency ; tx_power ; antenna_gain
14.23941117 ; 45.93641468 ; 5.0 ; 868.0        ; 10.0     ; -1.63
14.23755475 ; 45.93175757 ; 5.0 ; 868.0        ; 10.0     ; -1.63
14.23740066 ; 45.93465074 ; 5.0 ; 868.0        ; 10.0     ; -1.63
```

The following long-form input file *nodes-868* would produce the same results:

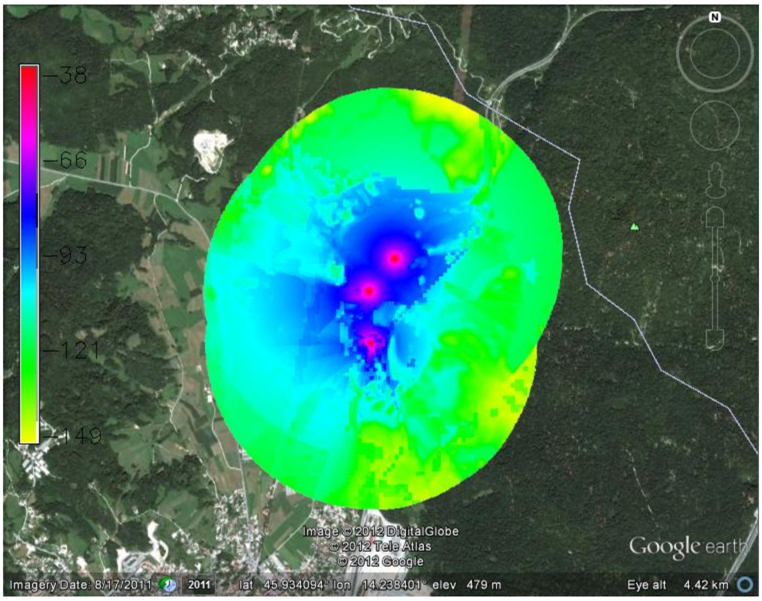
```
e           ; n           ; h   ; tx_frequency ; tx_power ; antenna_type       ; antenna_direction ; antenna_tilt
14.23941117 ; 45.93641468 ; 5.0 ; 868.0        ; 10.0     ; ANT-868-CW-HW-omni ; 0.0               ; 0.0
14.23755475 ; 45.93175757 ; 5.0 ; 868.0        ; 10.0     ; ANT-868-CW-HW-omni ; 0.0               ; 0.0
14.23740066 ; 45.93465074 ; 5.0 ; 868.0        ; 10.0     ; ANT-868-CW-HW-omni ; 0.0               ; 0.0
```

Example 1: Standard coverage map, radius 1 km

The following command is executed:

```
./wsn.coverage -i nodes-868-short -o results_868_1 -r 1
```

The resulting KMZ file *results_868_1.kmz* is displayed in Google Earth as shown in the following picture. The numbers in the legend (-149, -121, -66, -38) are shown in black, which is a bad choice for Google Earth (white would be fine), but a good one for display within the Web portal.

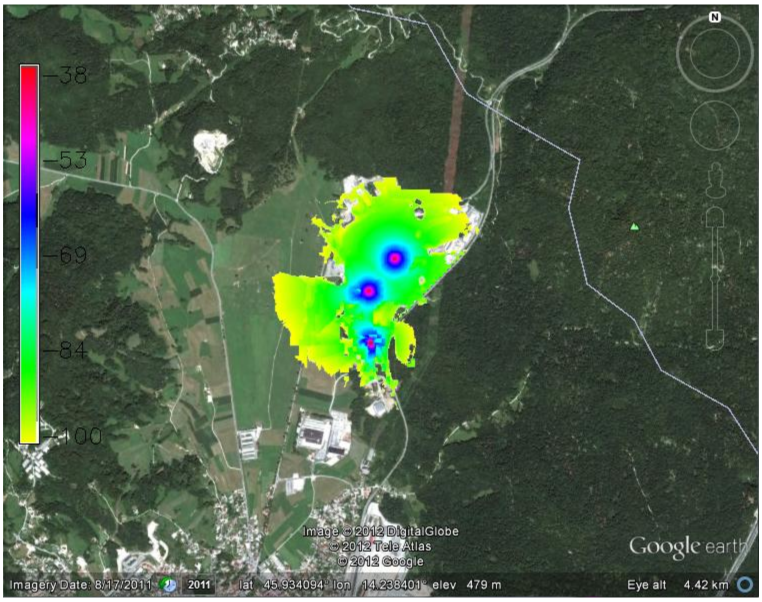


Example 2: Coverage map with rx_treshold -100 dBm, radius 1 km

The following command is executed:

```
./wsn.coverage -i nodes-868-short -o results_868_2 -r 1 --rx_threshold=100
```

The resulting KMZ file *results_868_2.kmz* is displayed in Google Earth as shown in the following picture. (The numbers in the legend are -100, -84, -53, -38.)

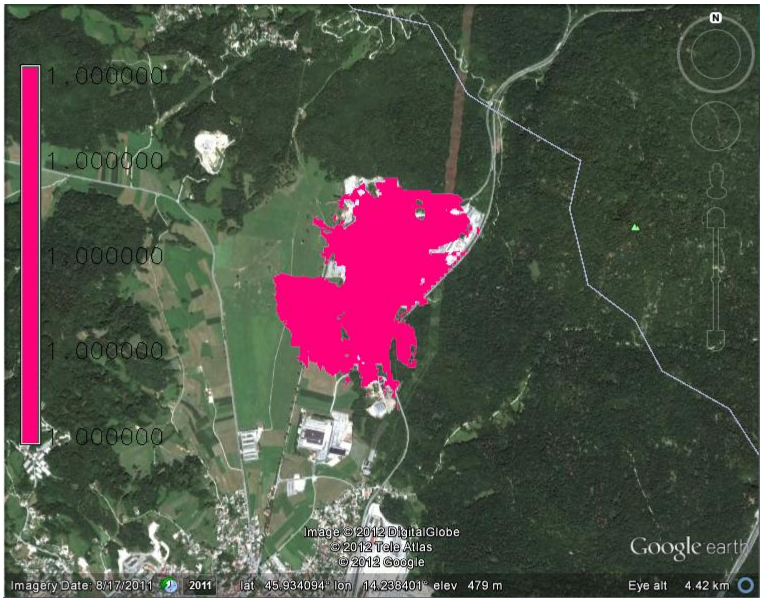


Example 2: Single-colored coverage map with rx_treshold -100 dBm, radius 1 km

The following command is executed:

```
./wsn.coverage -i nodes-868-short -o results_868_3 -r 1 --rx_threshold=100 -1
```

The resulting KMZ *file results_868_3.kmz* is displayed in Google Earth as shown in the following picture. (In this case the legend does not make any sense and only displays the value of 1.0.)



1.1.2 Receiver power computation (command *wsn.rxpowers*)

Web server requests coverage computation by using the following Linux command:

```
 wsn.rxpowers --input_tx=<tx_nodes_file> --input_rx=<rx_points_file> --output=<rx_powers_file> \
--radius=<radius_in_km> [--help]
```

or in short form:

```
wsn.rxpowers --input_tx=<tx_nodes_file> --input_rx=<rx_points_file> -o <rx_powers_file> \
-r <radius_in_km> [-h]
```

The command returns 0 upon successful completion or an error code otherwise. It may also print some descriptive information on stdout/stderr, which the Web portal can safely ignore.

Usage (an example)

The *input_tx* file *nodes-868-short*, which specifies the transmitting nodes, is the same as used in the examples for *wsn.coverage*:

```
e           ; n           ; h   ; tx_frequency ; tx_power ; antenna_gain
14.23941117 ; 45.93641468 ; 5.0 ; 868.0        ; 10.0     ; -1.63
14.23755475 ; 45.93175757 ; 5.0 ; 868.0        ; 10.0     ; -1.63
14.23740066 ; 45.93465074 ; 5.0 ; 868.0        ; 10.0     ; -1.63
```

The *input_rx* file *nodes_rx*, which specifies the receive locations, defines two locations, which are in these case identical to the locations of the first and third transmitting nodes:

```
14.23941117, 45.93641468
14.23740066, 45.93465074
```

The following command is executed:

```
./wsn.rxpowers --input_tx=nodes-868-short --input_rx=nodes-rx -o results_868_points -r 1
```

The resulting *output* file *results_868_points* is:

```
-37.8934124756,-101.102045898,-90.4912976074
-88.808062439,-93.1413220215,-39.6339482117
```

The first and the third value in the first and the second line respectively (37.9 and 39.6 dBi) represent identical transmit and receive locations and should be discarded.