

NV Editor (First Version)

User's Guide

List of contents

1) Overview	1
2) Panel Description	2
3) CSV Files	3
4) BRT Files	3
5) CNV Files	3
6) NVItemsDefine.def file	4
7) NVTypesDefine.def file	4
8) Functionality	6
a) Connection with mobile phone	6
b) Uploading/Downloading	6
c) Using BRT Files	6
d) Using CNV Files	7
e) Creating CSV Files	7
f) Sending SPC code to Mobile Phone	7
g) Editing NV Items.	8
h) Selecting NV Items.	8
i) Sorting NV Items	10
9) Active, not active and not accessible NV Items.	10
7) Adding new NV Items	10

1) Overview

NV Editor is a tool which enables editing NV Items content and storing results in files or in a mobile phone. Content of each NV Item is displayed in readable format in both decimal and hexagonal value.

NV Editor Functionality:

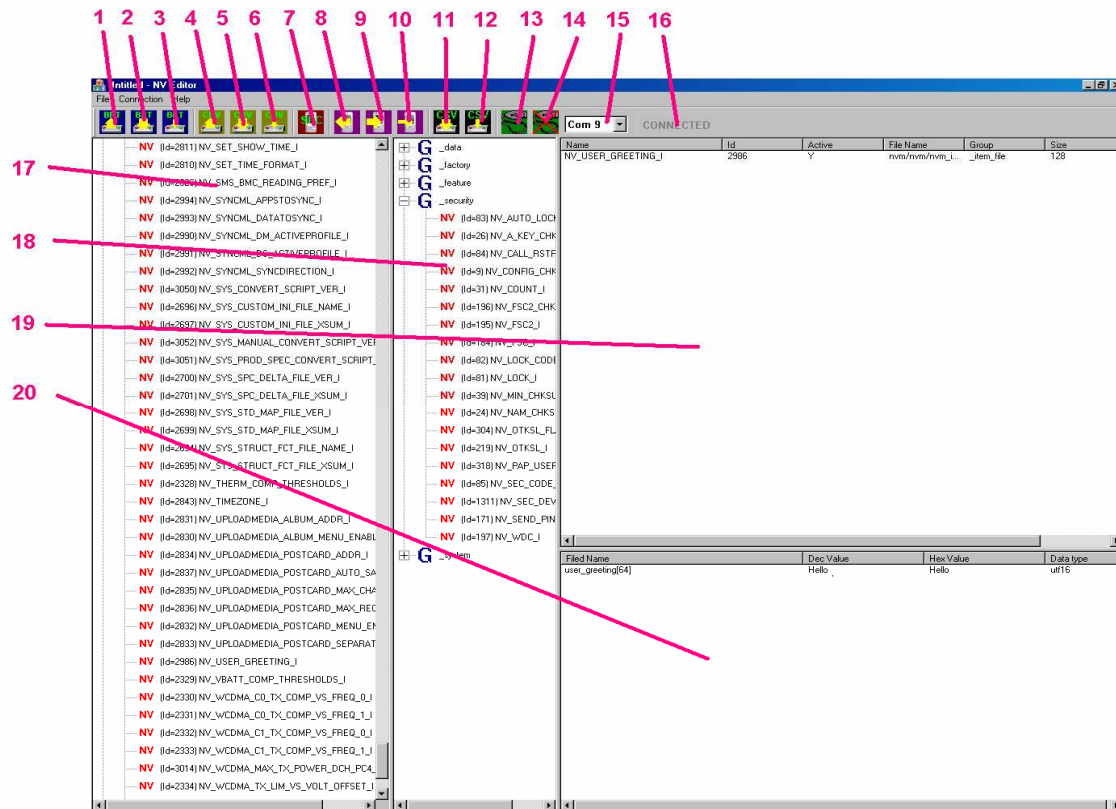
- Connecting/Disconnecting Mobile Phone
- Uploading NV Items from Mobile Phone
- Downloading NV Items to Mobile Phone
- Loading NV Items stored in BRT and CNV files.
- Saving all NV Items to BRT and CNV files.
- Generating CSV files
- Sending SPC code to mobile phone
- Editing NV Items' contents.
- GUI
 - i. Selecting NV Items for storing in files or mobile
 - ii. Sorting NV Items by Name, ID, Size, Data Type...
 - iii. Displaying NV Items' content in readable format in both a decimal and a hexagonal value

- iv. Editing NV Items' content in readable format in both decimal and hexagonal value

NOTE:

To use NV Editor QPST server has to be installed on the PC.

2) Panel Description



- 1 – Load NV Items from a BRT file
- 2 – Save all NV Items in a BRT file
- 3 – Save selected NV Items in a BRT file
- 4 – Load NV Items from a CNV file
- 5 – Save all NV Items in a CNV file
- 6 – Save selected NV Items in a CNV file
- 7 – Send SPC code to Mobile Phone
- 8 – Upload NV Items from Mobile
- 9 – Download all NV Items to Mobile
- 10 – Download selected NV Items to Mobile
- 11 – Create input CSV file with all NV Items
- 12 – Create input CSV file with selected NV Items
- 13 – Connect Server
- 14 – Disconnect Server
- 15 – Combo Box for choosing Com Port for connection with mobile
- 16 – Status of connection
- 17 – *Main list* with loaded NV Items
- 18 – *Selected list* with selected NV Items

- 19 – NV List view
- 20 – NV Content view

3) CSV Files.

CSV is a text file. This is an input file for uploading concrete NV Items from mobile. Only NV Items listed in CSV file will be uploaded from mobile. CSV files can be created by using some text editor or by Microsoft Excel (file have to be save with *.csv extension), or by NV Editor.

Format of CSV file is like that:

NV Name;NV id

NOTE:

The first line in file is not reading. This line can be used for comments for example.

Example:

NV Items – example file (comments)
NV_AAGPS_2G_MO_LRSUPPORT_I;1929
NV_AAGPS_3G_MO_LRSUPPORT_I;1960
NV_AAGPS_APP_TRACKING_GPSIDLE_THSLD_I;1923
NV_AAGPS_APP_TRACKING_GPSON_THSLD_I;1937

4) BRT File.

BRT is a binary file which can be an output or an input file. NV Items can be saved to this file or loaded from this file. BRT files in NV Editor have the same format like files using by the *Backup Restore Tool*.

5) CNV File

CNV is a binary file which can be an output or an input file. NV Items can be saved to this file or loaded from this file.

CNV can be used for downloading NV Items to mobile phone by using CEFS mechanism. To do it you have to do the following steps:

- create CNV file
- name of CNV file must be “nv_cnv.bin”
- crate EFS image
- copy nv_cnv.bin file to directory nvm/nvm/
- create CEFS
- download CEFS to mobile phone
- nv_cnv.bin will be automatically detected and all NV Items stored in this file will be automatically downloaded to phone

NOTE:

To create EFS image and CEFS you can use *EFs Simulator* tool for example.

6) NVItemsDefine.def File

NVItemsDefine.def describe all information about each NV Item like: *NV Name;Id;File Name;Group;Size;NV Data Type*

The most important information is: Id, Size and NV Data Type. Correct Id and Size are necessary for correct uploading and downloading from/to mobile phone.

Correct NV Data Type is necessary for correct conversion NV content from a row of bytes to readable format and in the opposite direction.

Format of this file is like this:

```
#NVItems for IB7
NV_BROWSER_TYPE_I;305;nvm/nvm/nvm_data;_data;0x0002;nvi_byte_type
NV_BTFD_OLPC_FLOOR_DB_I;939;nvm/nvm/nvm_data;_data;0x0002;nvi_int8_type
NV_BT_ADJUST_I;308;nvm/nvm/nvm_display;_display;0x0003;nvi_int2_type
```

NOTE:

If you want to write any comments use the sign ‘#’ in the beginning of line which is a comment.

7) NVTypesDefine.def File

NVTypesDefine.def file describes all NV Data Types. This file is necessary for correct conversion NV Item’s content to readable format.

NV Editor displays NV Content in readable format. It means that all data is converted from row of bytes reading from mobile to readable format. Readable format means that data is displayed as a one of the following “Simply Types”:

```
boolean;          /* Boolean value type. */
uint32;           /* Unsigned 32 bit value */
uint16;           /* Unsigned 16 bit value */
uint8;            /* Unsigned 8 bit value */
int32;            /* Signed 32 bit value */
int16;            /* Signed 16 bit value */
int8;             /* Signed 8 bit value */
byte;            /* Unsigned 8 bit value type. */
word;            /* Unsigned 16 bit value type. */
dword;           /* Unsigned 32 bit value type. */
uint1;           /* Unsigned 8 bit value type. */
uint2;           /* Unsigned 16 bit value type. */
uint4;           /* Unsigned 32 bit value type. */
int1;            /* Signed 8 bit value type. */
int2;            /* Signed 16 bit value type. */
int4;            /* Signed 32 bit value type. */
sint31;          /* Signed 32 bit value */
sint15;          /* Signed 16 bit value */
sint7;           /* Signed 8 bit value */
int64;           /* Signed 64 bit value */
uint64;          /* Unsigned 64 bit value */
qword[2];        /* Unsigned 64 bit value */
string[n];       /* ANSI String */
utf8[n];         /* UTF-8 String */
utf16[n];        /* UTF-16 String */
```

Most of NV Items are stored in structures. These structures contain types describing above or the others structures. To make a correct conversion knowledge of content of each structure is necessary. NVTypesDefine.def file describes all NV Types (structures) used to storing NV Items.

Format of NVTypeDefine.def is like that:

NVType Name; NV Type Name (or Simply Type) field Name; NV Type Name (or Simply Type) field Name;...

The first filed is *NVType name*. The next fields contain names of type of every structure's field and name of variable. To describe array you have to define number of elements in brackets after the name of variable (the same way like declaration variable in C/C++).

Example:

The below line:

nvi_err_log_type; byte err_count; byte file_name[8]; uint16 line_num; boolean fatal; this

is a definition of *nvi_err_log_type* which is a structure like:

```
typedef PACKED struct
{
    boolean          active;   NOT DESCRIBED NVTypeDefine.def FILE
    byte             err_count;
    byte             file_name[8];
    word             line_num;
    boolean          fatal
} nvi_err_log_type;
```

NOTE:

The first field in structure – ***boolean active*** is not used for storing data. So this field is not using for describing *NVType* in *NVTypeDefine.def*

If any field inside structure is a structure you have to describe this structure in *NVTypeDefine.def* file too.

Example 2:

If we have a structure:

```
typedef PACKED struct
{
    boolean          active;   NOT DESCRIBED IN NVTypeDefine.def FILE
    some_structure    Something[3];
} nvi_some_type;
```

We must know structure ***some_structure*** and describe it in *NVTypeDefine.def* file too.

If ***some_structure*** is for example like:

```
typedef PACKED struct
{
    byte             Field_1[3];
    int8             Field_2;
    boolean          Field_3;
} some_structure;
```

We must add two lines to *NVTypeDefnie.def*:


```
nvi_some_type;some_structure Something[3];  
some_structure;byte Field_1[3];int8 Field_2;Boolean Field_3;
```

During conversion name of each structure field is automatically built by using names from all “parents” structure names.

8) Functionality

a) Connecting with mobile phone

To connect phone do following steps.


- 1) Connect phone by USB to PC
- 2) Choose correct Com Port
- 3) Press Button  to connect Mobile Phone.
- 4) If connection is successful caption CONNECTED appears

To disconnect phone do following steps.


- 1) Press Button  to connect Mobile Phone.
- 2) Caption NOT CONNECTED appears

b) Uploading/Downloading NV Items


To upload NV Items from Mobile Phone do following steps:

- a) Phone has to be connected with PC via correct COM Port.
- b) Press Button  to upload NV Items form Mobile Phone.
- c) Choose some *.csv file which contains list of NV Items for uploading.
- d) After that *main list* will contain all NV Items uploaded from Mobile. *Selected list* will be empty. Previous content of both lists will be deleted.

To download all NV Items to Mobile Phone do following steps:


- 1) Phone have to be connected with PC via correct COM Port
- 2) Press Button  to download all NV Items from both lists – *main list* and *selected list*

To download only selected NV Items to Mobile Phone do following steps:

- 1) Phone have to be connected with PC via correct COM Port
- 2) Press Button  to Download only selected NV Items.
- 3) All NV Items from *selected list* will be downloaded to Mobile Phone.


c) Using BRT Files

To load NV Items from a BRT file do followings steps:


- a) Press Button  to load NV Items from file
- b) Choose correct file (*.brt)

- c) All NV Items stored in a BRT file will be displayed in *main list*. *Selected list* will be *empty*. Previous content will be deleted from both lists.

To save all NV Items to a BRT file do following steps:


- 1) Press Button  to save all NV Items to a BRT file.
- 2) Choose correct file or create new one.
- 3) After that all NV Items from both lists will be saved in a BRT file.
- 4) Content of both lists won't change.

To save only selected NV Items to a BRT file do following steps:


- 1) Press Button  to save only selected NV Items to a BRT file.
- 2) Choose correct file or create new one.
- 3) After that only NV Items from *selected list* will be saved in a BRT file.
- 4) Content of both lists won't change.

d) Using CNV files


To load NV Items from a CNV file do followings steps:

- 1) Press Button  to load NV Items from CNV file
- 2) Choose correct file (*.cnv)
- 3) All NV Items stored in a CNV file will be displayed in *main list*. *Selected list* will be *empty*. Previous content will be deleted from both lists.

To save all NV Items to a CNV file do following steps:


- 1) Press Button  to save all NV Items to a CNV file.
- 2) Choose correct file or create new one.
- 3) After that all NV Items from both lists will be saved in a CNV file.
- 4) Content of both lists won't change.

To save only selected NV Items to a CNV file do the following steps:


- 1) Press Button  to save only selected NV Items to CNV file.
- 2) Choose correct file or create new one.
- 3) After that only NV Items from *selected list* will be saved in a CNV file.
- 4) Content of both lists won't change.

e) Creating CSV Files

To create input CSV file with all NV Items do the followings steps:

- 1) Press Button  to create CSV file
- 2) Chose correct file or create new one
- 3) After that CSV file will be ready to use.

To create input CSV file by only selected NV Items do the followings steps:

- 1) Press Button  to create CSV file
- 2) Chose correct file or create new one
- 3) After that CSV file will be ready to use.

f) Sending SPC code to Mobile Phone

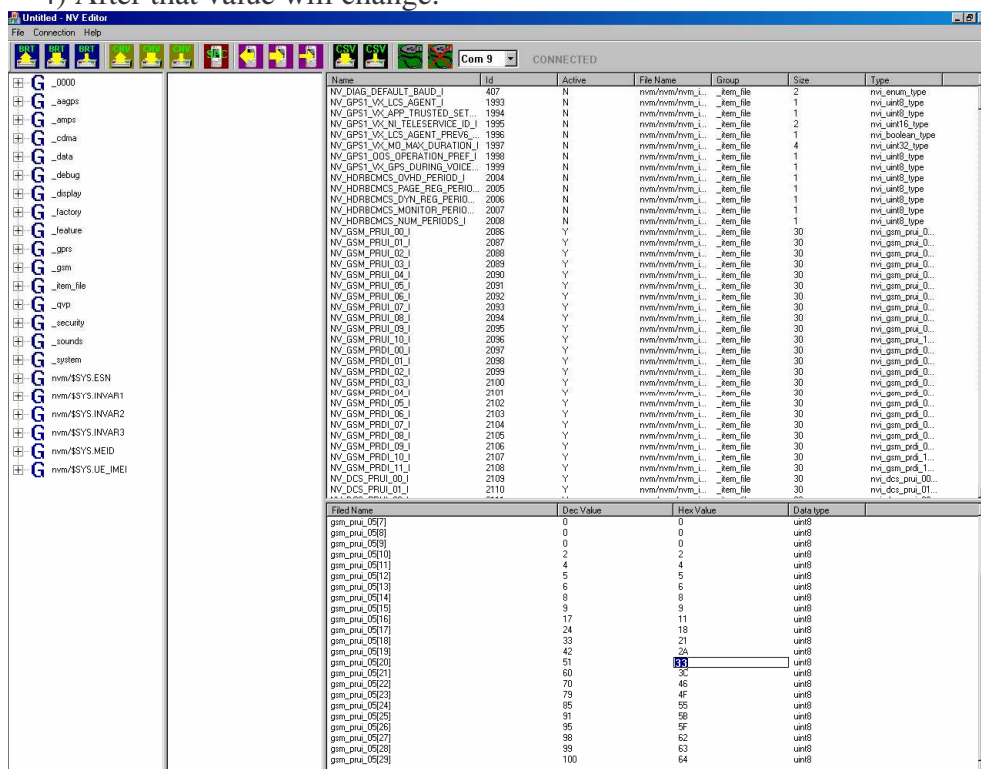
To send SPC code to Mobile Phone do the following steps:

- 1) Press Button to send SPC code
- 2) Write correct SPC code and press OK.
- 3) If the SPC code is not correct Mobile Phone will be locked for 10 second. In this period it is not possible to send or receive any information from Mobile Phone.

g) Editing NV Items.

To edit NV Item content do the following steps:

- 1) Choose concrete NV Item in main list or selected list or in list view for editing.
- 2) In content view double click on field you want to edit (decimal or hexagonal value).
- 3) Write new value and press ENTER or click on the other field.
- 4) After that value will change.

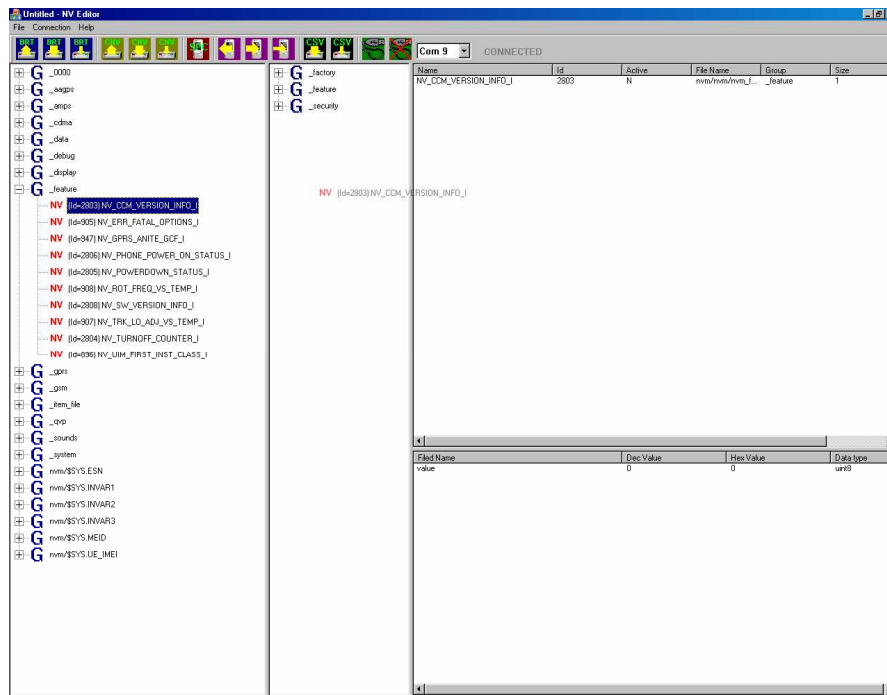


h) Selecting NV Items

NV Editor has a drag&drop mechanism for selecting NV Items.

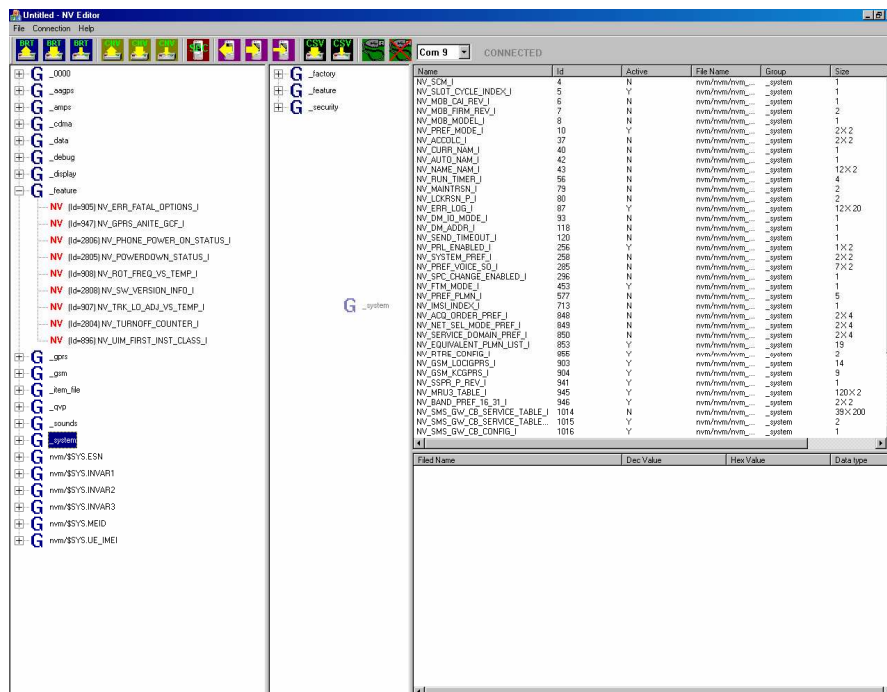
To select/unselect one NV Items do the following steps:

- 1) Choose NV Item in main or selected list and drag and drop it to the next list.
- 2) If NV Item is from not exist group in target list the new group appears in a target list
- 3) If NV item is the last Item in group in source list this group disappears from source list.



To select/unselect whole group of NV Items do the following steps:

- 1) Choose NV Item group in main or selected list and drag and drop it to the next list
- 2) If NV Item Group not exists in a target list it appears in a target list.
- 3) If NV Item Group exists in a target list this Group will be extended with new NV Items
- 4) NV Item Group disappears from source list



i) Sorting NV Items

To sort NV Items in NV List View click column for suitable sorting.

9) Active, not active and not accessible NV Items.

Some NV can't be uploaded from mobile. These NV Items are signed in NV Editor List view with field *Active* as X.

Some NV Items are uploaded from mobile but they aren't activated. To active them user have to edit this NV Items. To do it safely user should edit all fields of NV Item content, because not editing fields will be filled with zero value. Some times it can be dangerously for mobile.

During storing NV Items to file or downloading NV Items to mobile phone only *active* NV Items will be storing to file or downloading to mobile phone.

10) Enabling loading/uploading new NV Item's :

NV Editor enables working on NV Items which are described in *NVItemsDefine.def* file and with NV Data Types describing in *NVTypesDefine.def*. So if we want to have possibility to load/upload new NV Item using NV Editor we need to do two steps.

- 1) Modify *NVItemsDefine.def* file by adding new line with all information about NV Item:
- 2) Modify *NVTypesDefine.def* file by adding description for new NV Item type. If NV Item type was described earlier we don't need to describe this NV Item type the second time. If NV Item type contain some structure we need to describe this structure too (if they didn't be described earlier).