Version: 2012.08.28
Language: English

# CHECK.BIN EDITOR MANUAL


Tomáš Růžička
t_ruzicka@email.cz
www.djbozkosz.wz.cz


Typeset with LaTeX

# Contents

# Introduction

**Check.bin** - is a file that stores navigation network for the movement of various game objects. The network is made up by points and their connections. Contains navigation to:

- trams, trains, airplanes
- pedestrians
- AI to the orientation in an area (AI for enemies and panic)

Navigation are linearly related points. For example navigation for pedestrians forms path. But navigation for AI is a kind of grid, which helps to navigation.

Check.bin contains **only navigation**! Pedestrians, trams and other objects are stored in *scene2.bin* file (see Appendix).

This document summarizes only instructions how to create navigation. Algorithm, for example how AI searches the shortest path, it depends on the self study.

References:

`http://en.wikipedia.org/wiki/Dijkstra%27s_algorithmn`
`http://en.wikipedia.org/wiki/Shortest_path_problem`
`http://www.youtube.com/results?search_query=shortest+path`

# Chapter 1

# Preparation for use Check.bin Editor

The following conditions are recommended to fulfill for correct functionality:

- *Check.bin Editor* (CHE for short) - author: *zibob32*.
- *Mafia* game in version 1.0.
- Reduce resolution than is currently used in the operating system and disable fullscreen, both in the Mafia *Setup.exe*.
- Extract *missions* folder with *MafiaDataXtractor*.
- Copy all 8 models of points from *models* folder in the editor into *Mafia\models* folder - to visualize points in the game.
- Mission without sectors are recommended. Sectors can be deleted using the *Sector del* tool (please backup the original *scene.4ds* mission file). See Appendix.
- This script (below) insert into *scene2.bin* of same mission using *BScriptView 4 - 6* (menu *Insert* → *Other*) or *DCED 2*:

```
dim_flt 1
// press objectives (F1) to display navigation
label CHE
wait 1000
ctrl_read 0, OBJECTIVES
if flt[0] = 1, -1, CHE
cleardifferences
loaddifferences "CHED.chg"
goto CHE
```

# Chapter 2

# Points and links

## 2.1 Types of points


Figure 2.1: Mafia with visualized navigation points.

Following table summarizes the most important types of points:

| **Type in hex** - sizeof(unsigned short int); | **Description** |
|---|---|
| 01 00 | Pedestrians |
| 02 00 | AI (humans, cars) |
| 04 00 | Trams, trains, planes |
| 08 00 | Place of standing for trams |
| 10 00 | Special place for AI |

Other options for point types can be set in the bit flags of each point.

## 2.2 Types of links

Linked points have between themselves normally two links - one in each direction. This table contains the most important links:

| **Type in hex** - sizeof(unsigned short int); | **Description** |
|---|---|
| 01 00 | Pedestrians |
| 02 00 | AI (humans, cars) |
| 04 00 | Trams, trains, planes (forward direction) |
| 84 00 | Trams, trains, planes (backward direction) |
| 10 00 | Other |

For example point type 01 00 doesn't need to have same type of link (in this case 01 00). But eg trams 04 00 must have link type 04 00 and 84 00 to surrounding two points, to guarantee the **direction** where tram will be riding.

# Chapter 3

# Interface of editor

Below images are from the editor, complemented by complete translations.

## 3.1 Main window

The main window contains a list of all points that are stored in the file. The window is not used for the insertion points. Alternatively, after for additional adjustment of points.

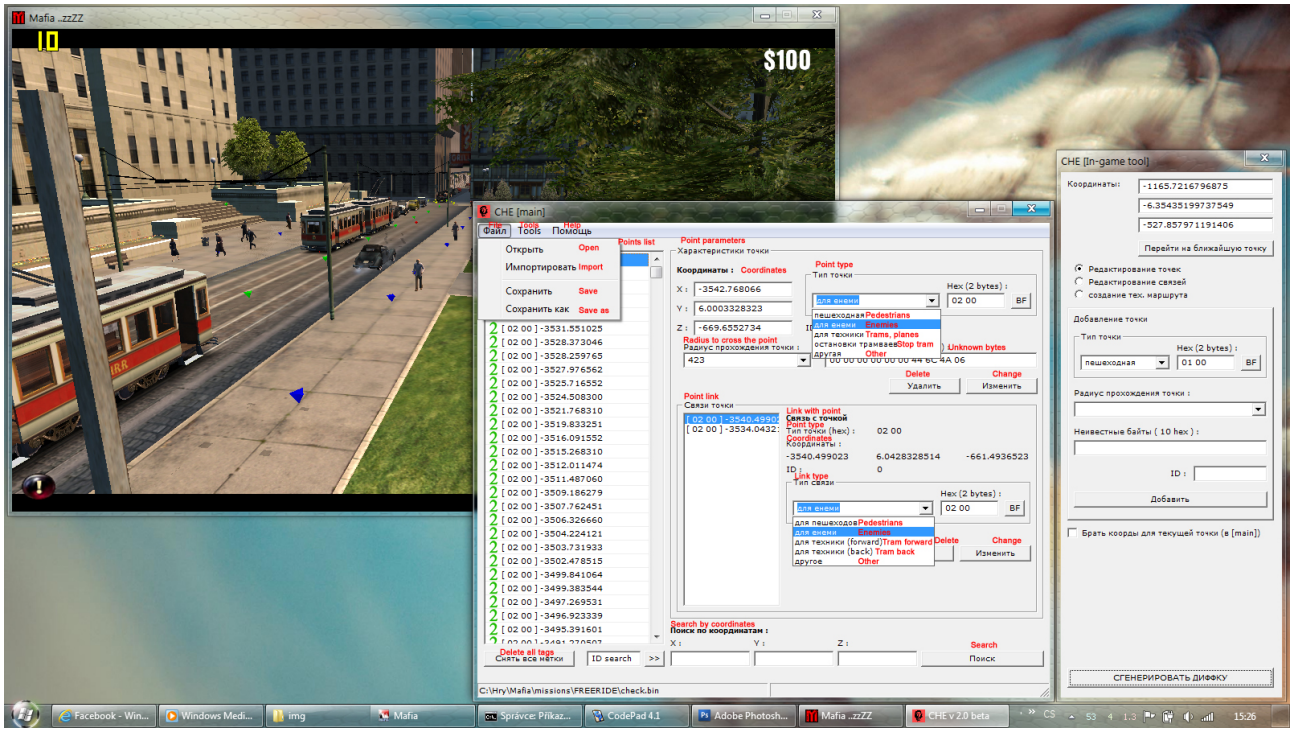



Figure 3.1: Mafia is appropriate to run in a window for easy switching between windows.

## 3.2 In-game tool

In-game tool is an important tool which is used for real-time getting position of player in the game (only for 1.0 version of game). The tool also allows on identified coordinates directly add new items, so the user is not impeded by debugging positions of points. Points can be also viewed in the game in fifty meters radius from the player.

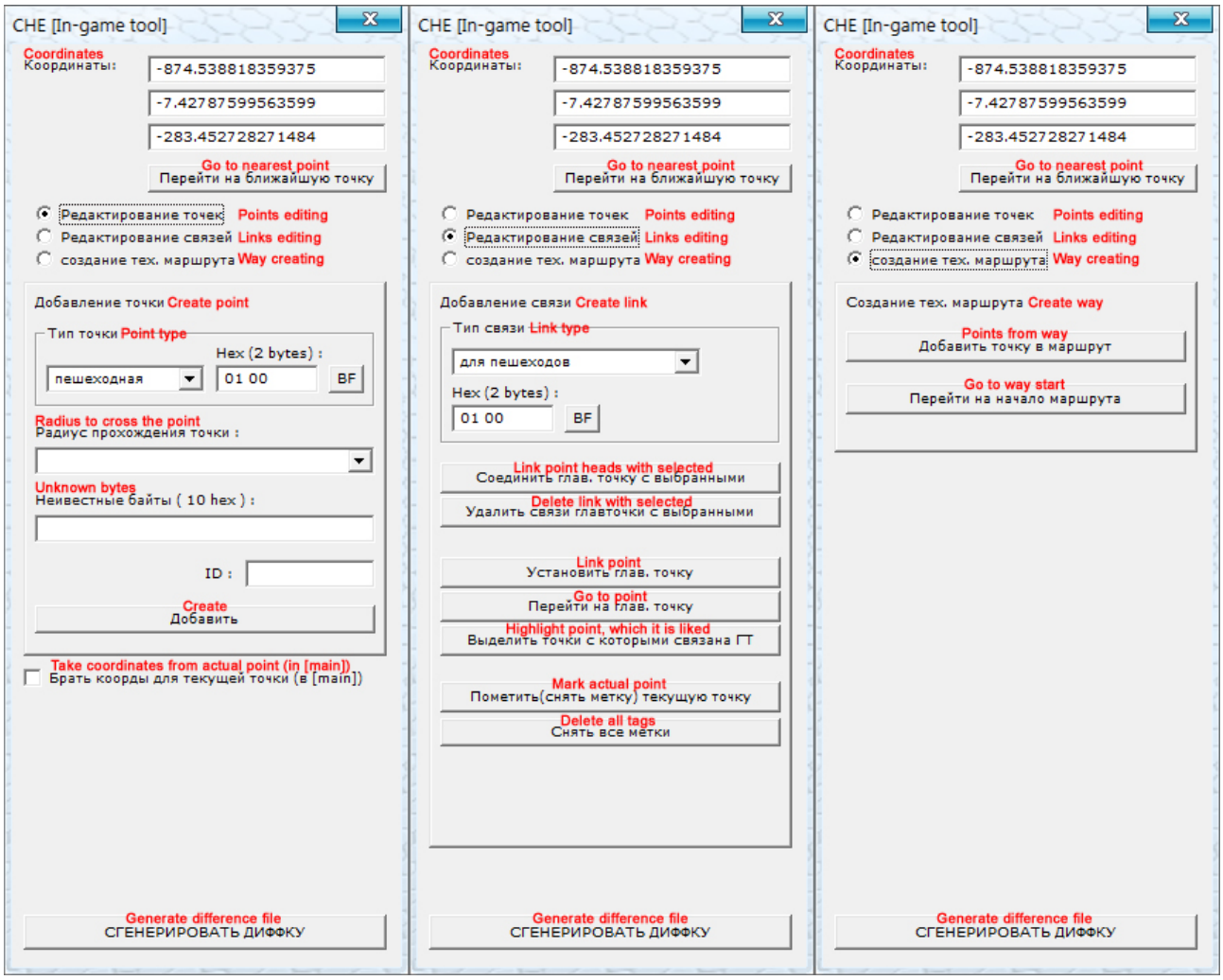



Figure 3.2: Total interface of in-game tool.

## 3.3 La Tierra

La Tierra draws points and their links to interactive view, which is primary designed to manage point links.

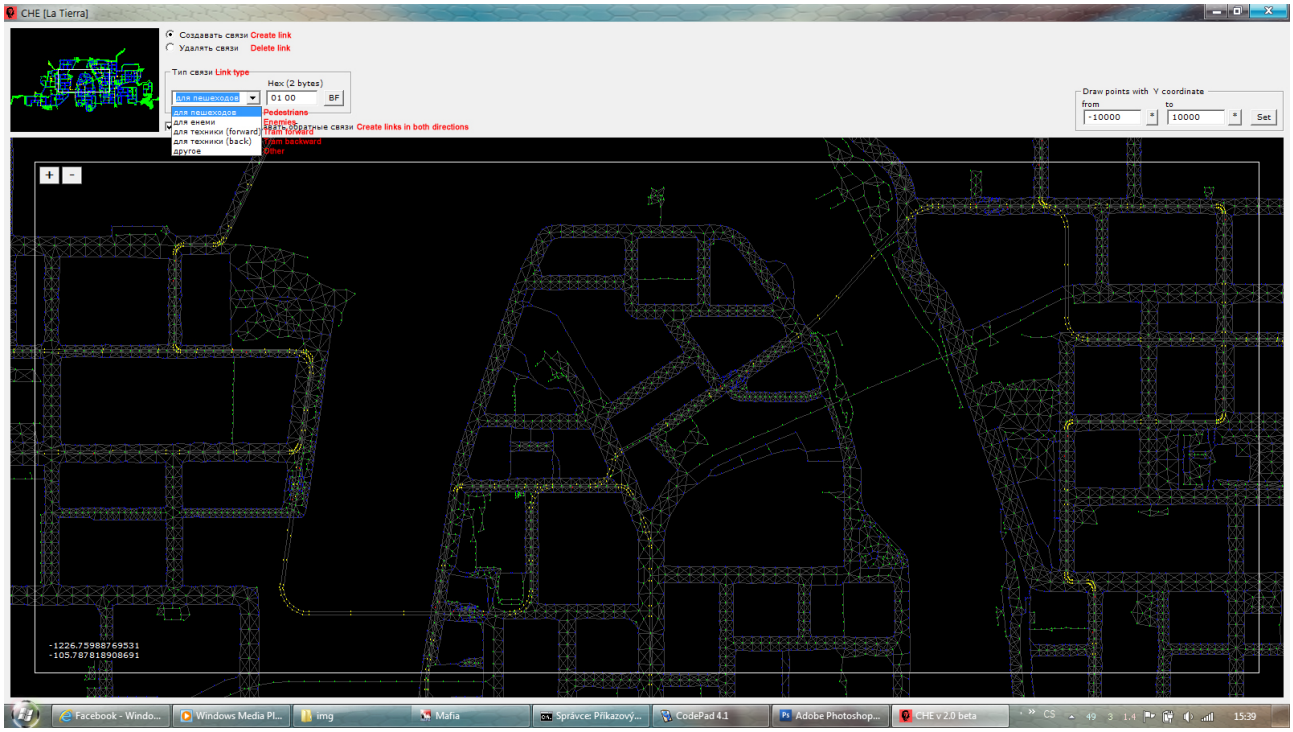



Figure 3.3: You can quickly create connections by displaying points in view.

# Chapter 4

# Using the editor

## 4.1 Preparation

- After loading required mission in Mafia please switch to operating system, run CHE and open (*File* → *Open*) *check.bin* file of equivalent mission.
- After opening mission in CHE, open *In-game tool* from *Tools* → *In-game mode*. *In-game tool* can be opened only if same mission is already loaded.
- If you move with player in the game you can see in *In-game tool* updating coordinates by actual position.
- To generate points click to *Generate difference file*. It creates *diff\CHED.chg* file that includes points near player. To show them in the game hit *F1* - objectives.

## 4.2 Creating points

- Place player in the desired location, where you want to create point.
- Set first option - *Points edititng* in the *In-game tool*.
- In box *Point type* select required type of point (see table above), then set *Radius to cross the point* to 50 by default, *Unknown bytes* are filled automatically by null bytes and set *ID* to zero (usually). This *ID* is same value as referenced *xx* in *enemy_move xx* script command. To add the point click to *Create* button.
- Repeat previous three steps to create required number of points in the mission area.
- Meanwhile, you should check location of points in the game by exporting a difference file and hitting *F1* key.

## 4.3 Creating links

- Point links is created in *La Tierra*. Open this tool in *Tools* → *La Tierra*.

- To move in preview click on thumbnail in the left top of window (faster) or click and hold in preview outside the white borders. You can also obstruct to draw points to height (Y coordination) by set the limit in top right corner of window.
- To create new links set first option *Create link*. In box *Link type* select required type of link (see table above) and check *Create links in both directions*.
- Click on first required point and then on second required point - this creates new links in both directions of points. To unlink mouse click on preview by right button of mouse.

## 4.4 Editing and deleting points and their links

- You can use main window for manage points and links. In points box click on *Change* - changed data are saved or click on *Delete* - point and his links are deleted. But if you delete point, shared links in linked points are changed to unknown. You have to delete these links too. Better way: delete links first.
- To delete links open *La Tierra* and select second option *Delete links*. At first, click on first point and then on second point - it deletes first link. To delete second link, click on same points in revesre order.
- Then you can safely delete points in main window.

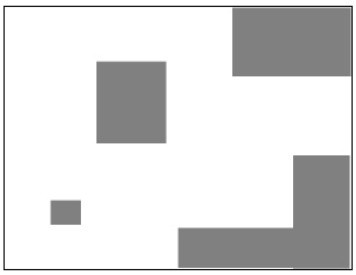
## 4.5 Saving

- To save file click on *File* → *Save*. Editor automatically backups older *check.bin* file to *BackUp* folder in the mission.

# Appendix A

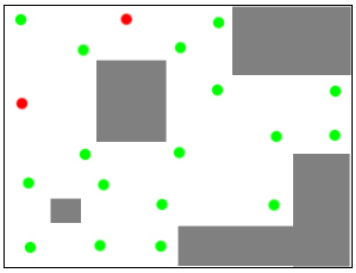
# Example of navigation

Following image sequence shows creating of points (type 02 00) and their links (type 02 00) in the interior, where creates navigation network for AI.



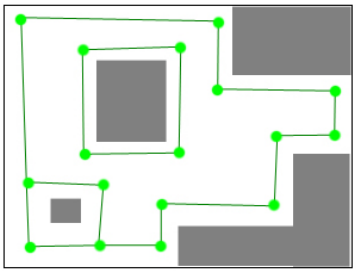
Room without navigation.

White surface illustrates an area available for the characters and gray surface illustrates an area unavailable for the characters - for example: solid interior equipment, desks, cabinets, columns, etc.

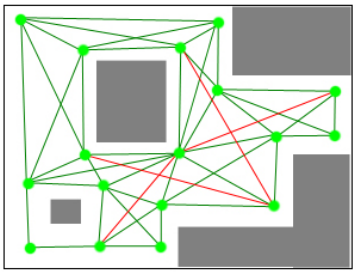


Room with points of type 02 00. Remember: if humans will be using the navigation, they will be walking only over the points and their links (except pedestrians). Place new points to outer and inner corners in the room, where characters can maneuver. Points should be given into plane only if the points are too far apart.

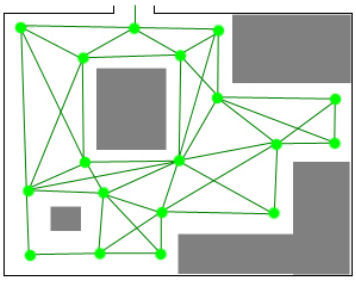
Red points are possibly insertable points, because neighboring points are away from each other.



The first step of linking (type 02 00) points (type 02 00).



The final links. While you creating a links, be careful to add very large number of links - on the picture are red. Maximum number of links from the point should be ± 8.



Finally, example how the navigation map change, if the door are in the room.

# Appendix B

# Erasing sectors

Exported points in the *CHG* file are visible only in the *Primary sector*. But problem is, if you need to visualize the points in the interior, whose rooms are in own visual sectors. Points cannot be seen in these sectors.

Sectors and other Zmodeler unsupported objects (glow objects, etc.) can be removed by copying the *4DS* file into the same folder as the program *sector_del.exe*. In this case it will be *scene.4ds* file from the mission. After run the application is the file automatically removed. To show details, run the application via the terminal.

Is strongly recommended to **backup** the original file, because the application it irreversibly change.

# Appendix C

# The definition and modification of objects in the scene2.bin file dependent on the check.bin file

## C.1 Pedestrians

To use navigation map by pedestrians, you need to insert them into *scene2.bin* file as object with type: point and object definition with equivalent name and with type:*Ped definition*. You can do this with *DCED 2* or *DNC Extractor*. Object of point type is standard point. Important is the definition of pedestrian and appropriate location of the point.

*DCED 2* allows to create definitions of pedestrians, but *DCED 2* very often damages the *scene2.bin* files and editing pedestrians through this program doesn't work correctly. Therefore, there is described more secure way.

In *DNC Extractor* at first load the mission that contains ped definition, eg *FREERIDE*. Then check required item in *Objects* list, eg *manhattan_people* and same item in *Objects definitions* list. Click to *Extract selected* program creates *DNC* files in *DNCes* folder in the folder where is *DNCextractor.exe*. Reopen the application and load required mission. Then click to *Import*, select *DNC* files and save the file.

**Caution:** if the mission was before saved in *Mafia World Editor*, you need to open mission in this editor and delete (shortcut *Del*) **point** called *Primary Sector* (if this point exist). The point have a same name as standard sector, called *Primary sector*. You can identify this point: if you select a sector called *Primary Sector* everything will be red, but if you select point called *Primary Sector* scene will be same.

Then save the mission. If the **point** isn't in the mission, just resave the mission.

Inserted object of point contains a coordinations that determines place to generate pedestrians. To place point you can use *BScriptView* or *Mafia World Editor*.

Extended properties are stored in object definition that you can safely edit with any hex editor. Follow image describes an important values of pedestrians in definition:

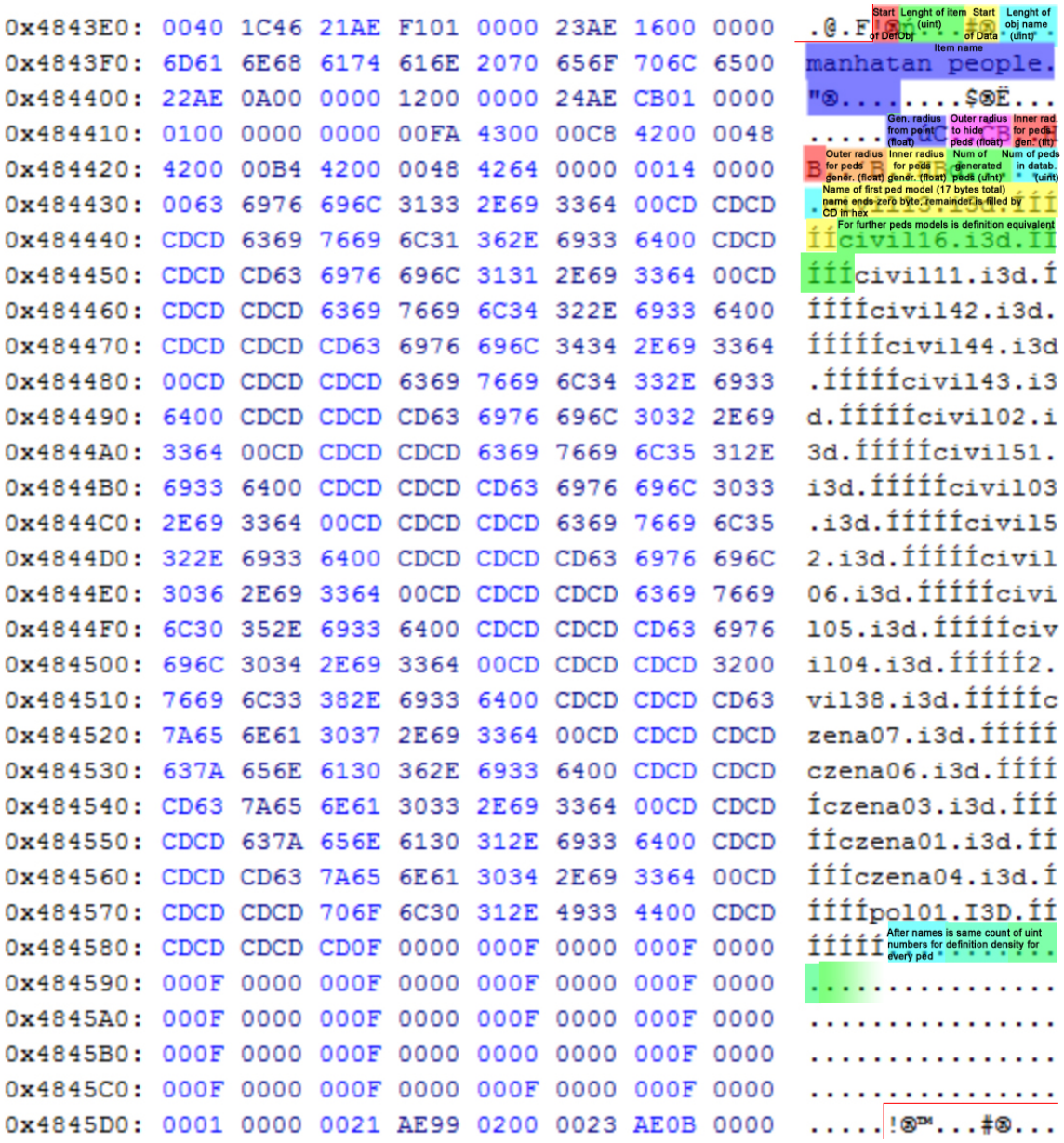



Figure C.1: Analysis of pedestrian object definition in scene2.bin file.

Generation of peds is determined too by position of player and an area which is specified by inner and outer radius from player: *purple, red, orange, yellow - everything is float*.

On the navigation map can be generated number of peds specified by: *green - unsigned int*.

Below in file is defined: number of humans in database: *cyan - unsigned int* and names of hunams. String with name of human model is ended by zero byte and remainder is filled with *0xCD* bytes. For every name is reserved space 17 bytes. Number of hunams is defined before database.

At the end is stored ratio density of every item from database: *cyan - unsigned int*.

When you are editing this file with hex editor you **mustn't change length** of file.

But you can change length of object definition in DNC file, but then you have to recalculate length of item in at start of file.

## C.2 Trams and trains

You can get model object of tram and his definition with *DNC Extractor* described in the previous chapter.

You have to place the wagon directly to train navigation route. Extended properties are stored in object definition that you can safely edit with any hex editor. Follow image describes an important values of tram in definition:

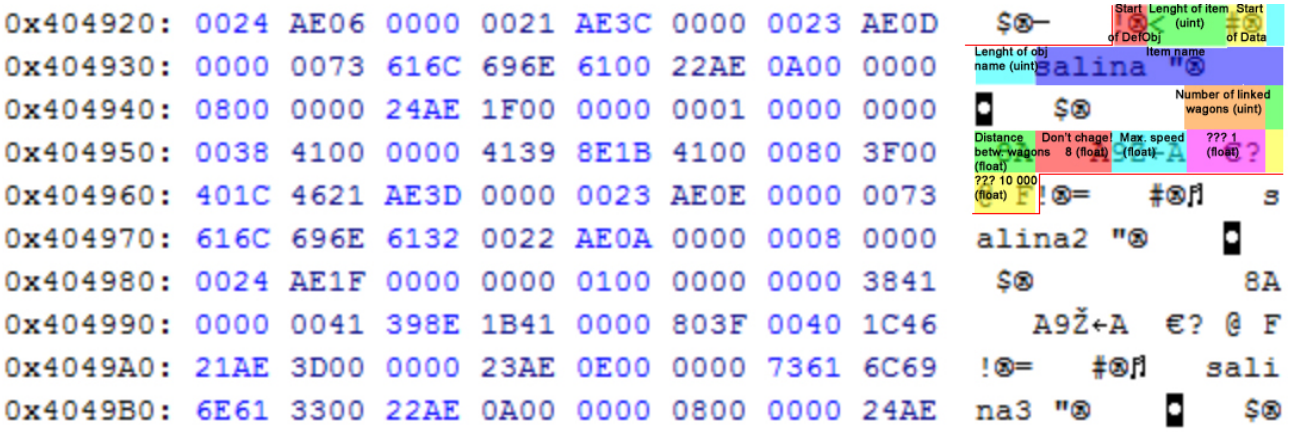



Figure C.2: Analysis of tram object definition in scene2.bin file.

Important values are: number of linked wagons, distance between wagons and maximum speed. This speed is reached only if the navgation points are far from each other.

When you are editing this file with hex editor you **mustn't change length** of file.

But you can change length of object definition in DNC file, but then you have to recalculate length of item in at start of file.

# Appendix D

# Converting numbers

Unsigned int is stored in hex form. Float is real number stored according to standard *IEEE 754* in form: sign * 2 pow exponent * mantissa.

Numbers are usually stored from least significant byte to most significant byte. It means, when you are converting numbers you have to reverse the order of bytes.

For example - integer: decimal form: *22 384* hex form: *00 00 56 12* stored form: *12 56 00 00*.

For example - real number: decimal form: *15.256* hex form: *41 74 18 93* stored form: *93 18 74 41*.

Integer numbers you can convert in calculator.

Real numbers you can convert with online convertor: `http://www.h-schmidt.net/FloatConverter/IEEE754.html` or in program *Base Converter* and or directly in hex editor *Hex Workshop*.

If you are using *Base Converter* you have to set the byte order to *Intel* and data type to *float (32)*. Then you directly copy quaternions of bytes (without reversing the order) from hex editor to converter.