



RGV1  
Basics of modding  
**EMPIRE EARTH II**



*Basics of modding Empire Earth 2.*

*Asmop RGV1, creator of more than 10 main mods for the game EE2. Founder of the Empire Earth 4 project - a global mod for the game EE2.*

*Co-authors*

*P.M.A.*

*Michael34*

*All texts belong to their respective authors.*

RGV1



# Content:

<b>Introduction</b>	<b>4</b>
<b>1How to change the main characteristics of the unit.</b>	<b>5</b>
<b>2How to add a new unit and building.</b>	<b>9</b>
<b>3What can be changed in the UnitType block</b>	<b>17</b>
<b>3.1 RPS List</b>	<b>22</b>
<b>3.2 List of effects</b>	<b>23</b>
<b>4 What can be changed in the UnitModel block</b>	<b>42</b>
<b>5 How to add a new icon to the game.</b>	<b>44</b>
<b>6 How to add a new 3D vehicle or building model to the game. Basics of working in the NifSkope program.</b>	<b>50</b>
<b>7 How to add a new 3D human or animal model to the game.</b>	<b>89</b>
<b>8 How to add player color bars.</b>	<b>108</b>
<b>9 How to add seasonal textures.</b>	<b>118</b>
<b>10 How to add new textures for terrain cover, mountains, cliffs, river beds and oceans.</b>	<b>122</b>
<b>11 How to add new introductory videos and change videos in the menu. How to change the menu background and loading screens.</b>	<b>126</b>
<b>12 Mods for Empire Earth II. Modders Empire Earth II</b>	<b>129</b>
<b>Conclusion</b>	<b>130</b>

## Introduction:

All text books are about The Art of Supremacy, but are also valid for EE2.  
In some cases, colored underlining of the terms was used:

**Blue bold** - chapters

**Blue** - examples and meanings, subsections.

**Red** - main block or main parameter.

**Pink** - subblocks or main block values (parameter).

Most of the game's modifiable parameters include:

1 description,

2 examples,

3 accepted values

The sign (?) Indicates questions that are not fully developed.

The style of writing is completely different as it was developed by different authors.  
However, the material is readily available and is enough to master the basics of EE2  
modding.



# 1. How to change the main characteristics of the unit.

It explains how to change the basic parameters of units such as health, attack, speed, etc. In the latest versions of the game, all the main parameters are located in three files:

upgrade\_unittypes.csv

upgrade\_unittypes\_EE2X.csv (only EE2 AoS has this file)

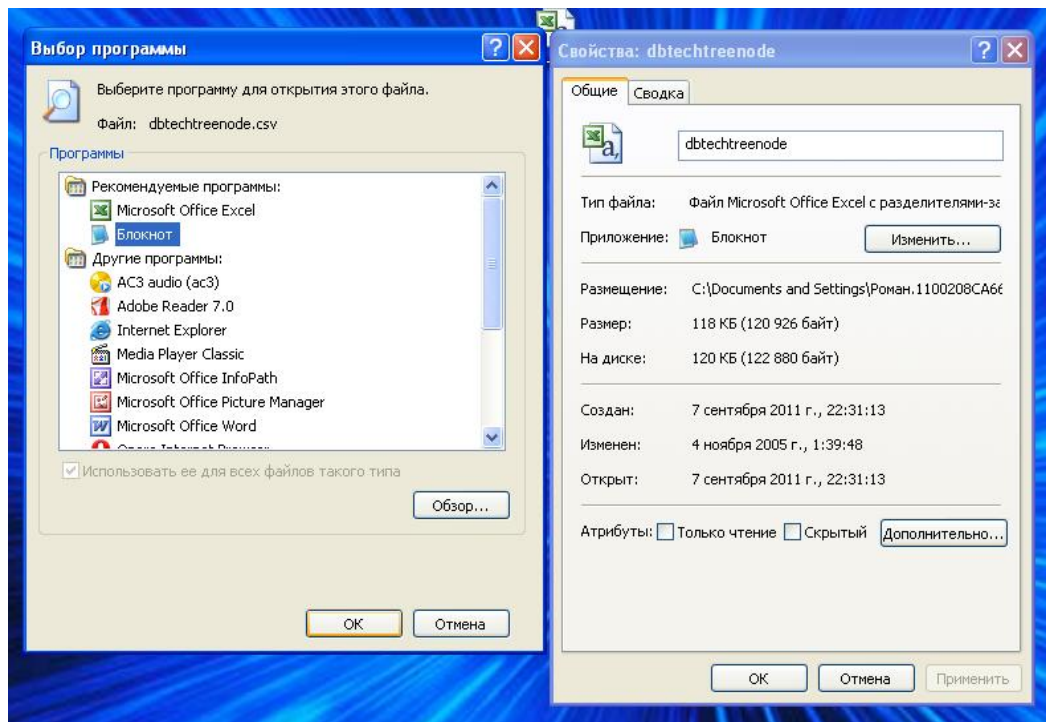
dbtechreenode.csv

All these files are in the folder TechTree

Is in:

C:\Program Files\Sierra\Empire Earth II\zips\_ee2x\EE2X\_db.zip\EE2X\_db\TechTree

These files can be easily opened with a notepad. To make it easier to open, please follow the steps below. Copy (drag) any csv file from the archive to the desktop. Match: open with notebook. Select apply to all file types. Now all csv files in the archive will be double clicked to open in notepad.



File upgrade\_unittypes.csv



In upgrade\_unittypes.csv (upgrade\_unittypes\_EE2X.csv - everything is the same) there is a row for each unit, for example:

```
LightInfantryUpgradeEpoch1,LightInfantry,"""Unit""",1,60,7,7,6,3,37,30,20,0,0,10,0,0,0,tx_utn_LightInf_e1_name,tx_utn_LightInf_e1_pname,LightInfantry_01,,icon_unit_bowman,[0 0 0 0],0,0,vtt_unit_bowman,[v_LightInfantry_01_sel01],[v_LightInfantry_01_com01 v_LightInfantry_01_com02],[v_LightInfantry_01_com01 v_LightInfantry_01_com02 v_LightInfantry_01_spe01],[],[LightInfantry_Improve_Reset LightInfantryEpoch1Attack],All
```

Various parameters are separated by commas, all numerical parameters can be changed at your discretion. When saving, always select "save changes".

Actually, I am giving the values of all parameters from left to right:

1 LightInfantryUpgradeEpoch1 - unit name, in this case for light infantry that is available in the first era

2 LightInfantry - unit type, in this case light infantry

3 ""Unit"" - belonging to a unit or building

4 The era from which the individual comes EPOCH

5 Number of life HP

6 Unit view range LOS. However, there is a limit to this parameter in the game. LOS cannot be greater than 15

7 Unit attack power DAMAGE

8 RANGE A unit's attack range can be values other than total, for example 0,8

9 Charging time RELOAD

10 Unit creation time BUILDTIME

11 The amount of food needed to create a unit FOOD

12 The amount of wood needed to create a unit WOOD

13 The amount of stone needed to create a unit STONE

14 The amount of gold needed to create a unit GOLD

15 The amount of tin needed to create a unit TIN

16 The amount of iron needed to create a unit IRON

17 The amount of saltpeter needed to create a unit SALTPETER

18 The amount of oil needed to create a unit OIL

19 The amount of uranium needed to create a unit URANIUM

20 tx\_utn\_LightInf\_e1\_name unit name displayed in the game

21 tx\_utn\_LightInf\_e1\_pname unit name (plural) displayed in the game

22 LightInfantry\_01 unit name for the computer AI player

23 the parameter is omitted

24 icon\_unit\_bowman unit icon

25 [0 0 0 0] internal game parameter

26 0 there should be a parameter specifying the size of the population the unit occupies. However, this parameter is defined directly in the unit text block.

27 0 parameter in the game

28 vtt\_unit\_bowman text tooltip for the unit.

29 [v\_LightInfantry\_01\_sel01] unit selection sound

30 [v\_LightInfantry\_01\_com01 v\_LightInfantry\_01\_com02] unit movement sounds (this sound commands it to move to another location)

31 [v\_LightInfantry\_01\_com01 v\_LightInfantry\_01\_com02 v\_LightInfantry\_01\_spe01] attack sounds

32 The sound of a dying individual

34 The unit belongs to a civilization. All - means that the unit can be produced by all civilizations.  
Korean - Only for the Korea civilization.

Greek  
Roman  
English  
American  
German  
Egyptian  
Turkish  
Babylonian  
Chinese  
Japanese  
Korean  
Inca

```
dblechtreeode - Блокнот
Файл Правка Формат Вид Справка

// NAME, DISPLAYNAME, TOOLTIP, EPOCH, SLOT, BRANCH, ICON, MENU, ROW, COLUMN, HOST, PRODUCE, UPGRADE, TIME, FOOD, WOOD, STONE, GOLD, TIN, IRON, SALTPETER, OIL, URANIUM, TECHPTS, TT
EPOCH 1
=====
// UNITS AND UPGRADES.....
Citizen, text_Citizen_name, 1,0,0,,0,1,0,CityCenter,Citizen,25,50,0,0,0,0,0,0,0,0,0,0, [],(None)
FishingShip, text_FishingShip_name, 1,0,,0,1,0,Dock,FishingShip,FishingShipUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
HeavyInfantry1, text_HeavyInfantry1_01_name, 1,0,,0,1,1,Barracks,HeavyInfantry1,HeavyInfantry1UpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
LightArtillery1, text_LightArtillery1_01_name, 1,0,,0,1,0,Mill,LightArtillery1,LightArtillery1UpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
LightInfantry, text_LightInfantry_01_name, 1,0,,0,1,0,Barracks,LightInfantry,LightInfantryUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
MerchantShip, text_MerchantShip_name, 1,0,,0,1,1,Dock,MerchantShip,MerchantShipUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Priest, text_Priest_name, 1,0,,0,1,0,Temple,Priest,PriestUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Scout, text_Scout_name, 1,0,,0,1,1,CityCenter,Scout,ScoutUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Spy, text_Spy_name, 1,0,,0,1,0,University,Spy,SpyUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Wargalley, text_Wargalley_01_name, 1,0,,0,2,0,Dock,Wargalley,WargalleyUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)

// UNIT UPGRADES.....
LightInfantry_01_Veteran, text_tt_upgrade_veteran, 1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,LightInfantry, LightInfantryEpochVeteran,60,66,66,0,0,66
LightInfantry_01_Elite, text_tt_upgrade_elite, 1,0,Improvement,icon_tech_upgrade_elite,0,1,3,LightInfantry, LightInfantryEpochElite,60,66,66,0,0,66,0,0,66
HeavyInfantry1_01_Veteran, text_tt_upgrade_veteran, 1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,HeavyInfantry1, HeavyInfantry1EpochVeteran,60,66,0,0,66
HeavyInfantry1_01_Elite, text_tt_upgrade_elite, 1,0,Improvement,icon_tech_upgrade_elite,0,1,3,HeavyInfantry1, HeavyInfantry1EpochElite,60,66,0,0,66,0,0,66
LightArtillery1_01_Veteran, text_tt_upgrade_veteran, 1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,LightArtillery1, LightArtillery1EpochVeteran,60,133,0,0,133
LightArtillery1_01_Elite, text_tt_upgrade_elite, 1,0,Improvement,icon_tech_upgrade_elite,0,1,3,LightArtillery1, LightArtillery1EpochElite,60,133,0,0,133,133
wargalley_01_veteran, text_tt_upgrade_veteran, 1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,wargalley, wargalleyepochveteran,80,0,100,0,0,100,0,0,0,0,0,0,[warga]
wargalley_01Elite, text_tt_upgrade_elite, 1,0,Improvement,icon_tech_upgrade_elite,0,1,3,wargalley, wargalleyepochelite,80,0,100,0,0,100,0,0,0,0,0,0,[warga]

// BUILDINGS.....
Barracks, text_Barracks_name, 1,0,,,2,1,0,Citizen,Barracks,BarracksUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Bridge, text_Bridge_name, 1,0,,0,1,0,2,Citizen,Bridge,BridgeUpgradeEpoch,20,0,0,25,25,0,0,0,0,0,0,0,0, [],(None)
CityCenter, text_CityCenter_name, 1,0,,0,1,1,0,Citizen,CityCenter,CityCenterUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Dock, text_Dock_name, 1,0,,2,1,2,Citizen,Dock,DockUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Fortress, text_Fortress_name, 1,0,,2,1,0,Citizen,Fortress,FortressUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
House, text_House_name, 1,0,,0,1,1,Citizen,House,HouseUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
Mill, text_Mill_name, 1,0,,0,2,1,Citizen,Mill,MillUpgradeEpoch,0,0,0,0,0,0,0,0,0,0,0,0, [],(None)
```

Citizen,text\_Citizen\_name,,1,0,,,0,1,0,CityCenter,Citizen,,25,50,0,0,0,0,0,0,0,0,0,,[],(None)

The main purpose of this file is to determine which unit - where is it produced, in which building - by whom. Various parameters are separated by commas, all numerical parameters can be changed at your discretion. When saving, always select "save changes"

Actually, I am giving the values of all parameters from left to right:

- 1 Citizen name of the unit (or building)
- 2 text\_Citizen\_name unit name used in the game, especially in game statistics
- 3 game parameters TOOLTIP (like the name of the research)
- 4 the era from which the new unit originates
- 5 Slot (used for research)
- 6 the parameter is used for research
- 7 however, the icon is not indicated in this file because it is already registered in upgrade\_unittypes.csv (the icon is indicated only for research)
- 8 0 - menu. By default, 0 is the buildings game menu, 1 is the civilian construction menu for workers, 2 is the military building construction menu for workers, respectively.
- 9 1 - stands for a row (line), the lowest row is 0, the next highest is 1, etc. There are a total of four rows numbered from 0 to 3.
- 10 0 - means the column (column), leftmost column is number 0, next column on the right is 1, etc. A total of four columns numbered from 0 to 3
- 11 the building in which the unit is produced (or a unit that can build a building)
- 12 once again the unit (or building) that is being produced.
- 13 unit upgrade.
- 14 production time of the unit
- 15 The amount of food needed to create a unit FOOD
- 16 The amount of wood needed to create a unit WOOD
- 17 The amount of stone needed to create a unit STONE
- 18 The amount of gold needed to create a unit GOLD
- 19 The amount of tin needed to create a unit TIN
- 20 The amount of iron needed to create a unit IRON
- 21 The amount of saltpeter needed to create a unit SALTPETER
- 22 The amount of oil needed to create a unit OIL
- 23 The amount of uranium needed to create a unit URANIUM
- 24 The number of points needed to create a unit
- 25 belonging to a civilization, you can specify a country or culture (for example, Middle East)
- 26 Research without which these studies cannot be carried out
- 27 Special labels

If some parameters are omitted, it means they are already registered in upgrade\_unittypes.csv. Parameters such as speed and pop are defined in UnitType blocks. More on this in the next chapter.



## 2 How to add a new unit and building.

Part 1. Understanding the game, creating a new unit in place of the old one.

Here is the first tutorial published on Igromania's website.

(The information provided relates to the original EE2. The supplement does the same.)

[http://www.igromania.ru/articles/47500/Empire\\_Earth\\_II.htm](http://www.igromania.ru/articles/47500/Empire_Earth_II.htm)

(the address of the article may change, look for the password „Autopsy Empire Earth II”)

Let's go deep into the game catalog. We are interested in its largest subdirectory - \ zips. Contains game archives which, as the folder name suggests, are in zip format. Thanks to this, every modern archiver can work with them. Let's get acquainted with the archives themselves. The db.zip archive contains many files with game parameters, fonts.zip stores the graphical representation of the game fonts, graphics.zip - models of units and buildings, hdrs.zip - scripts, sounds.zip - sound files in mp3 format (the archive contains the \sounds\Ambients , \ sounds\ Effects and \sounds\Vo, the first of which stores sounds emitted by the environment, the second - sound effects, the third - speech sound), textures.zip — bmp and tga textures. Both the sound element of the game and the graphic element (only textures) can be easily improved as commonly used formats are used to store this data. The most valuable archive for us is — db.zip. Most of the files in its directories have the extension ddf. To work with them, you just need a regular Windows "Notepad". In addition, it also includes a number of important csv files. They contain tables. Such files can be edited using Microsoft Excel. Let's go back to ddf and talk about their structure. Each such file is divided into thematic sections, designed as follows:

```
N {  
X = Y  
...  
}
```

N here — block name, X — parameter name from this section, Y — the importance of it.

Ellipses are usually replaced with more attributes and their values.

A block may contain subsections. They combine features with a similar purpose and are formed as described. In addition, related parameters can be combined into small groups. The latter have the same structure as file sections, but square brackets are used instead of curly braces in this case.

Let's move from theory to practice:

1. In Empire Earth II, we will replace the American scout 1-5 epochs with a Russian lancer.

To insert the lancer into the game, open the UniqueUnit\_American.ddf file (originally used to store the unique characteristics of American warriors) from the \ db \ Units \ directory in the db.zip archive. In the first section of the file - UnitType UniqueUnitAmerican1 - change the value of the parent attribute to HeavyInfantry2. After that, our unit will in many ways look like a representative of heavy infantry (and not a scout as it was with the first unique American unit) for the first five eras.

Then see the properties section. Change the parameter values in it SizeX (sets the fighter's height) and SizeY (unit width) to 0.2 and 0.175 respectively. Now the unit has slightly different dimensions. Outwardly, it will be larger than the warrior whose model we will take as a basis.

Time to adjust the combat parameters. In the group of abilities features, in the Attack subsection, assign the parameters range (distance), reloadTime (reload time) and damage (damage) to values 0.3, 1.8 and 8, respectively. In the Move group (movement parameter), reduce the value of the speed attribute to 2, a angSpeed (angular speed) and angAccel (angular acceleration) - to 360. This speed reduction is due to the heavy armament of the unit - a type of pike. "Weakness" will affect the vigilance of the individual. The group's view range indicator is responsible for this LOS, the value of which should be reduced to 6 - after all, our recruit specializes in attack, not reconnaissance.

The next step is to edit the appearance of the unit. As already mentioned, it will be based on the selector model. Copy the entire contents of the section UnitModel HeavyInfantry2\_05 (except the name, but with the trailing curly brace) file heavy\_infantry2.ddf to the UnitModel UniqueUnitAmerican1 block above (let me remind you that it is in the UniqueUnit\_American.ddf file). In this case, delete the original contents of the latter - leave only the section name.

Let's change the name of the unit. To do this, look in the \db\Text directory and find the file dbtext\_techtreenames.utf8 (contains the real names of all units, i.e. those that you can see during the game). Open it in Notepad and look for two more lines, one of which starts with a combination tx\_utn\_UUAmerican1\_name (used for singular unit name), druga — z tx\_utn\_UUAmerican1\_pname (plural name). Next to the first there is a word in quotation marks "Frontiersman" - It should be replaced with a word "Lancer". Correct the text in quotation marks from the second line to "Lancers".

It is also advisable to include information about the new unit in the description. In the dbtext\_unittypetips.utf8 file from the \ db \ Text folder in the line beginning with the combination it\_UUAmerican1 , correct the text in quotation marks. In its simplest form, an updated tooltip may appear like this: "The Lancer is armed with a long push weapon and can only be created by the Russians in the eras 1-5". You can come up with a more detailed description - it depends on the changes made.

Part 2. Create a new unit from scratch

If you understood the first part, great. You've practically mastered the basics of modding the game. Now let's try to create a completely new unit that can be played in a single player game (skirmish).

Let's create, for example, an assault squad of the 12th era, modeled on the Italian Arditi. The main feature of the new unit is the armament of the machine gun and an increased life parameter. I'll show you how it is done with the example of the AoS add-on, in the original EE2 everything is done the same way.

First, open the file:

C:\Program Files (x86)\Sierra\Empire Earth II\zips\_ee2x\EE2X\_db.zip\EE2X\_db\Units\UniqueUnit\_Roman.ddf. Find the blocks related to the unit there:UniqueUnitRoman3

```
UniqueUnit_Roman — Блокнот
Файл Правка Формат Вид Справка

//-----
// UNIQUE UNIT ROMAN ARDITI (EPOCHS 11-15)
//-----
unitType UniqueUnitRoman3
{
    parent = HeavyInfantry1
    properties {
        DisplayName = tx_utn_UURoman3_name
        DisplayNameScenedit = tx_utn_UURoman3_sname
        VerboseTooltip = vtt_unit_UURoman3
        HitPoints = 0
    }
    sounds {
    }
    abilities = [
        Attack { damage = 0; range = 0; reloadtime = 666; applyDamageTime = 0.5; weaponType = LightProjectile }
        ConvertAttack { convertTimeModifier = 3.0; range = 2.0; reloadtime = 4.0; applyDamageTime = 1.0; capture = 1 }
        LOS { range = 0 }
        Move { speed = 1.4375; angSpeed = 720 }
    ]
}

//-----
// UNIQUE UNIT IMPROVEMENTS
//-----

// Veteran upgrade
upgradeUnitTypeImprove UniqueUnitRoman3Veteran
{
    unitType = UniqueUnitRoman3
    name = tx_utn_veteran_modify_name
    upgradeRefs = [
        UniqueUnit_Improve_5
    ]
}

// Elite upgrade
upgradeUnitTypeImprove UniqueUnitRoman3Elite
{
    unitType = UniqueUnitRoman3
    name = tx_utn_elite_modify_name
    upgradeRefs = [
        UniqueUnit_Improve_5
    ]
}

//-----
// UNIQUE UNIT MODEL
//-----
unitModel UniqueUnitRoman3
{
    Parent = BaseHuman
    DefaultModel = CIV12_Arditi_Roman.nif
    childNames = [ UnitShadow Beam_Tracer01 Muzzle_Rifleman ]
    States {
        {
            StateName = Idle AnimName = CIV12_Arditi_Roman_idle.kf
            AnimVariants = [ CIV12_Arditi_Roman_idle2.kf ]
            AnimVariantsWeights = [ 0.2 ]
        }
        {
            StateName = walk AnimName = CIV12_Arditi_Roman_walk.kf
            StateName = CaptureAttack1 AnimName = CIV12_Arditi_Roman_capture.kf
            AnimVariants = [ CIV12_Arditi_Roman_capture2.kf ]
            AnimVariantsWeights = [ 0.3 ]
        }
        {
            StateName = Run AnimName = CIV12_Arditi_Roman_jog.kf
            StateName = Attack1 AnimName = CIV12_Arditi_Roman_attack.kf
            AnimVariants = [ CIV12_Arditi_Roman_attack2.kf ]
            AnimVariantsWeights = [ 0.2 ]
        }
    }
    TextKeys = [

```

Let's create a blank notepad file. Copy the block there UnitType UniqueUnitRoman3 and UnitModel UniqueUnitRoman3

Watch out for the parentheses! All opening and closing parentheses must be copied. You should have this:

```
Безымянный — Блокнот
Файл Правка Формат Вид Справка

unitType UniqueUnitRoman3
{
    parent = HeavyInfantry1
    properties {
        DisplayName = tx_utn_UURoman3_name
        DisplayNamescenedit = tx_utn_UURoman3_sname
        VerboseTooltip = vtt_unit_UURoman3
        HitPoints = 0
    }
    sounds {
    }
    abilities = [
        Attack { damage = 0; range = 0; reloadTime = 666; applyDamageTime = 0.5; weaponType = LightProjectile }
        ConvertAttack { convertTimeModifier = 3.0; range = 2.0; reloadTime = 4.0; applyDamageTime = 1.0; capture = 1 }
        LOS { range = 0 }
        Move { speed = 1.4375; angSpeed = 720 }
    ]
}

unitModel UniqueUnitRoman3
{
    Parent = BaseHuman
    DefaultModel = CIV12_Arditi_Roman.nif
    ChildNames = [ UnitShadow Beam_Tracer01 Muzzle_Rifleman ]
    States {
        {
            StateName = Idle AnimName = CIV12_Arditi_Roman_idle.kf
            AnimVariants = [ CIV12_Arditi_Roman_idle2.kf ]
            AnimVariantsWeights = [ 0.2 ]
        }
        {
            StateName = Walk AnimName = CIV12_Arditi_Roman_walk.kf
            StateName = CaptureAttack1 AnimName = CIV12_Arditi_Roman_capture.kf
            AnimVariants = [ CIV12_Arditi_Roman_capture2.kf ]
            AnimVariantsWeights = [ 0.3 ]
        }
        {
            StateName = Run AnimName = CIV12_Arditi_Roman_jog.kf
            StateName = Attack1 AnimName = CIV12_Arditi_Roman_attack.kf
            AnimVariants = [ CIV12_Arditi_Roman_attack2.kf ]
            AnimVariantsWeights = [ 0.2 ]
        }
        {
            TextKeys = [
                { Name = PLAY_SOUND Data = fx_machine_gun_fire KeyTime = 0.5 }
            ]
        }
        {
            StateName = Death AnimName = CIV12_Arditi_Roman_death.kf
            StateName = Flinch AnimName = CIV12_Arditi_Roman_protect01.kf
            StateName = BeingThrown AnimName = CIV12_Arditi_Roman_flying01.kf
            AnimVariants = [ CIV12_Arditi_Roman_flying02.kf ] AnimVariantsWeights = [ 0.5 ]
        }
        {
            StateName = ThrownDeath AnimName = CIV12_Arditi_Roman_ground_death01.kf
        }
    }
}

//
//
```

Now, for example, open the file:

C:\Program Files (x86)\Sierra\Empire Earth II\zips\_ee2x\EE2X\_db.zip\EE2X\_db\Units\american\_campaign\_units.ddf

And insert the contents of the notebook at the very end of the open file.



```

}
unitModel PresidentialHumvee
{
    ScaleType      = kScaleVehicle
    DefaultModel    = 11a13_preshumvee.nif
    DefaultState    = Idle
    States = [
        { StateName= Idle AnimName = 11a13_preshumvee_idle.kf
          TextKeys = [ { Name = STOP_CONTROLLER Data = NiuvController KeyTime = 0 }
                      ]
        }
        { StateName= walk AnimName = 11a13_preshumvee_walk.kf
          TextKeys = [ { Name = START_CONTROLLER Data = NiuvController KeyTime = 0 }
                      { Name = START_SOUND_UNTIL_ANIM_CHANGE Data = fx_move_halftrack KeyTime = 0 }
                      ]
        }
        { StateName= Death }
    ]
    childNames      = [ TheHealthBar SelectCircleMedium FX_ExhaustSmoke1 TankTracks ExplSelectorTank01 Preshumvee_FlyingParts ]
}

unitType UniqueUnitRoman3
{
    parent = HeavyInfantry1
    properties {
        DisplayName = tx_utn_UURoman3_name
        DisplayNameScenEdit = tx_utn_UURoman3_sname
        VerboseTooltip = vtt_unit_UURoman3
        HitPoints = 0
    }
    sounds {
    }
    abilities = [
        Attack { damage = 0; range = 0; reloadTime = 666; applyDamageTime = 0.5; weaponType = LightProjectile }
        ConvertAttack { convertTimeModifier = 3.0; range = 2.0; reloadTime = 4.0; applyDamageTime = 1.0; capture = 1 }
        LOS { range = 0 }
        Move { speed = 1.4375; angSpeed = 720 }
    ]
}

unitModel UniqueUnitRoman3
{
    Parent          = BaseHuman
    DefaultModel    = CIV12_Arditi_Roman.nif
    ChildNames      = [ UnitShadow Beam_Tracer01 Muzzle_Rifleman ]
    States = [
        { StateName = Idle AnimName = CIV12_Arditi_Roman_idle.kf
          AnimVariants = [ CIV12_Arditi_Roman_idle2.kf ]
          AnimVariantsWeights = [ 0.2 ]
        }
        { StateName = walk AnimName = CIV12_Arditi_Roman_walk.kf
          StateName = CaptureAttack1 AnimName = CIV12_Arditi_Roman_capture.kf
          AnimVariants = [ CIV12_Arditi_Roman_capture2.kf ]
          AnimVariantsWeights = [ 0.3 ]
        }
        { StateName = Run AnimName = CIV12_Arditi_Roman_jog.kf
          StateName = Attack1 AnimName = CIV12_Arditi_Roman_attack.kf
          AnimVariants = [ CIV12_Arditi_Roman_attack2.kf ]
          AnimVariantsWeights = [ 0.2 ]
        }
        { TextKeys = [ { Name = PLAY_SOUND Data = fx_machine_gun_fire KeyTime = 0.5 } ]
        }
        { StateName = Death AnimName = CIV12_Arditi_Roman_death.kf
          StateName = Flinch AnimName = CIV12_Arditi_Roman_protect01.kf
          StateName = BeingThrown AnimName = CIV12_Arditi_Roman_flying01.kf
          AnimVariants = [ CIV12_Arditi_Roman_flying02.kf ] AnimVariantsWeights = [ 0.5 ]
        }
    ]
}

```

Now let's go directly to the block editing:

We need to set a new name,

We change:

UnitType UniqueUnitRoman3 on UnitType PanzerGrenadier

UnitModel UniqueUnitRoman3 on UnitModel PanzerGrenadier

It is also necessary to change, as in the previous case:

DisplayName = tx\_utn\_UURoman3\_name Nazwa jednostki

VerboseTooltip = vtt\_unit\_UURoman3 Opis jednostki

DisplayNameScenEdit = tx\_utn\_UURoman3\_sname - This line can be completely removed.

In this case, the name will be used in the map editor unit DisplayName.

HitPoints = 150 - We set our hit points parameter.

We also change other parameters. As a result, we get:

```
american_campaign_units — Блокнот
Файл Правка Формат Вид Справка

ScaleType = kScaleVehicle
DefaultModel = l1a13_preshumvee.nif
DefaultState = Idle
States = [
    { StateName= Idle AnimName = l1a13_preshumvee_idle.kf
      TextKeys = [ { Name = STOP_CONTROLLER Data = NiuvController KeyTime = 0 } ]
    }
    { StateName= Walk AnimName = l1a13_preshumvee_walk.kf
      TextKeys = [ { Name = START_CONTROLLER Data = NiuvController KeyTime = 0 }
                  { Name = START_SOUND_UNTIL_ANIM_CHANGE Data = fx_move_halftrack KeyTime = 0 } ]
    }
    { StateName= Death }
]
ChildNames = [ TheHealthBar selectCircleMedium FX_ExhaustSmoke1 TankTracks ExplSelectorTank01 Preshumvee_FlyingParts ]
}

unitType PanzerGrenadier
{
    parent = HeavyInfantry1
    properties {
        DisplayName = tx_utm_UURoman3_name
        DisplayNameScenedit = tx_utm_UURoman3_sname
        verboseTooltip = vtt_unit_UURoman3
        HitPoints = 150
    }
    sounds {
    }
    abilities = [
        Attack { damage = 10; range = 0.5; reloadTime = 0.5; applyDamageTime = 0.5; weaponType = LightProjectile }
        ConvertAttack { convertTimeModifier = 3.0; range = 2.0; reloadTime = 4.0; applyDamageTime = 1.0; capture = 1 }
        LOS { range = 15 }
        Move { speed = 1.4375; angSpeed = 720 }
    ]
}

unitModel PanzerGrenadier
{
    Parent = BaseHuman
    DefaultModel = CIV12_Arditi_Roman.nif
    ChildNames = [ UnitShadow BeamTracer01 Muzzle_Rifleman ]
    States = [
        { StateName = Idle AnimName = CIV12_Arditi_Roman_idle.kf
          AnimVariants = [ CIV12_Arditi_Roman_idle2.kf ]
          AnimVariantsWeights = [ 0.2 ]
        }
        { StateName = Walk AnimName = CIV12_Arditi_Roman_walk.kf
          StateName = CaptureAttack1 AnimName = CIV12_Arditi_Roman_capture.kf
          AnimVariants = [ CIV12_Arditi_Roman_capture2.kf ]
          AnimVariantsWeights = [ 0.3 ]
        }
        { StateName = Run AnimName = CIV12_Arditi_Roman_jog.kf
          StateName = Attack1 AnimName = CIV12_Arditi_Roman_attack.kf
          AnimVariants = [ CIV12_Arditi_Roman_attack2.kf ]
          AnimVariantsWeights = [ 0.2 ]
        }
        { Name = PLAY_SOUND Data = fx_machine_gun_fire KeyTime = 0.5 }
    ]
    { StateName = Death AnimName = CIV12_Arditi_Roman_death.kf }
    { StateName = Flinch AnimName = CIV12_Arditi_Roman_protect01.kf }
    { StateName = BeingThrown AnimName = CIV12_Arditi_Roman_flying01.kf
      AnimVariants = [ CIV12_Arditi_Roman_flying02.kf ] AnimVariantsWeights = [ 0.5 ]
    }
    { StateName = ThrownDeath AnimName = CIV12_Arditi_Roman_ground_death01.kf }
    ]
}

//
//
```

Save changes to the file american\_campaign\_units.ddf

I saved without giving a new name to the unit.

If we did everything right, we will find two Arditì units in the map editor. One of them is our new unit.



Now let's add the possibility of producing him in the 12th era.

Let's open the file EE2X\_db.zip\EE2X\_db\TechTree\dbtechreenode.csv

Find a string in it:

UniqueUnitRoman3,text\_UURoman3\_name,,11,0,,0,1,3,Barracks,UniqueUnitRoman3,UU\_Roman3UpgradeEpoch11,0,0,0,0,0,0,0,0,0,0,,[],(UniqueUnit)

Copy it and paste it at the very top of the document. Let's change it as follows:

PanzerGrenadier,text\_UURoman3\_name,,12,0,,0,2,3,Barracks,PanzerGrenadier,,20,20,20,0,0,0,0,0,0,0,,[],(None)

We receive:

```
dbtechreenode - Блокнот
Файл Правка Формат Вид Справка
// NAME, DISPLAYNAME, TOOL TIP, EPOCH, SLOT, BRANCH, ICON, MENU, ROW, COLUMN, HOST, PRODUCE, UPGRADE, TIME, FOOD, WOOD, STONE, GOLD, TIN, IRON, SALTPETER, OIL, URANIUM, TECHPTS, TTCTI
// UNITS AND UPGRADES,.....
PanzerGrenadier,text_UURoman3_name,,12,0,,0,2,3,Barracks,PanzerGrenadier,,20,20,20,0,0,0,0,0,0,0,,[],(None)

Citizen,text_Citizen_name,,1,0,,0,1,0,CityCenter,Citizen,,25,50,0,0,0,0,0,0,0,0,0,,[],(None)
FishingShip,text_FishingShip_name,,1,0,,0,1,0,Dock,FishingShip,FishingShipUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
HeavyInfantry1,text_HeavyInfantry1_01_name,,1,0,,0,1,1,Barracks,HeavyInfantry1,HeavyInfantry1UpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
LightArtillery1,text_LightArtillery1_01_name,,1,0,,0,1,0,Mill,LightArtillery1,LightArtillery1UpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
LightInfantry1,text_LightInfantry1_01_name,,1,0,,0,1,0,Barracks,LightInfantry1,LightInfantry1UpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
MerchantShip,text_MerchantShip_name,,1,0,,0,1,1,Dock,MerchantShip,MerchantShipUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Priest,text_Priest_name,,1,0,,0,1,0,Temple,Priest,PriestUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Scout,text_Scout_name,,1,0,,0,1,1,CityCenter,Scout,ScoutUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Spy,text_Spy_name,,1,0,,0,1,0,University,Spy,SpyUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
WarGalleys,text_WarGalleys_01_name,,1,0,,0,2,0,Dock,WarGalleys,WarGalleysUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)

// UNIT UPGRADES,.....
LightInfantry1_Veteran,text_tt_upgrade_veteran,,1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,LightInfantry,LightInfantryEpoch1Veteran,60,66,66,0,0,66,0
LightInfantry1_Elite,text_tt_upgrade_elite,,1,0,Improvement,icon_tech_upgrade_elite,0,1,3,LightInfantry,LightInfantryEpoch1Elite,60,66,66,0,0,66,0,0,0,0,0
HeavyInfantry1_Veteran,text_tt_upgrade_veteran,,1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,HeavyInfantry1,HeavyInfantry1Epoch1Veteran,60,66,66,0,0,66,0
HeavyInfantry1_Elite,text_tt_upgrade_elite,,1,0,Improvement,icon_tech_upgrade_elite,0,1,3,HeavyInfantry1,HeavyInfantry1Epoch1Elite,60,66,66,0,0,66,0,0,0,0,0
LightArtillery1_Veteran,text_tt_upgrade_veteran,,1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,LightArtillery1,LightArtillery1Epoch1Veteran,60,133,0,0,133,133
LightArtillery1_Elite,text_tt_upgrade_elite,,1,0,Improvement,icon_tech_upgrade_elite,0,1,3,LightArtillery1,LightArtillery1Epoch1Elite,60,133,0,0,133,133
WarGalleys_Veteran,text_tt_upgrade_veteran,,1,0,Improvement,icon_tech_upgrade_veteran,0,1,3,WarGalleys,WarGalleysEpoch1Veteran,80,0,100,0,0,100,0,0,0,0,0,0,0
WarGalleys_Elite,text_tt_upgrade_elite,,1,0,Improvement,icon_tech_upgrade_elite,0,1,3,WarGalleys,WarGalleysEpoch1Elite,80,0,100,0,0,100,0,0,0,0,0,0,0

// BUILDINGS,.....
Barracks,text_Barracks_name,,1,0,,2,1,0,Citizen,Barracks,BarracksUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Bridge,text_Bridge_name,,1,0,,1,0,2,Citizen,Bridge,BridgeUpgradeEpoch1,20,0,0,25,0,0,0,0,0,0,0,,[],(None)
CityCenter,text_CityCenter_name,,1,0,,1,1,0,Citizen,CityCenter,CityCenterUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Dock,text_Dock_name,,1,0,,2,1,2,Citizen,Dock,DockUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Fortress,text_Fortress_name,,1,0,,2,2,0,Citizen,Fortress,FortressUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
House,text_House_name,,1,0,,1,1,1,Citizen,House,HouseUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Mill,text_Mill_name,,1,0,,2,1,1,Citizen,Mill,MillUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Outpost,text_Outpost_name,,1,0,,2,0,1,Citizen,Outpost,OutpostUpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Outpost2,text_Outpost2_name,,1,0,,2,1,1,Citizen,Outpost2,Outpost2UpgradeEpoch1,0,0,0,0,0,0,0,0,0,0,,[],(None)
Palisade,text_Palisade_name,,1,0,,2,0,2,Citizen,Palisade,PalisadeUpgradeEpoch1,7,0,10,0,0,0,0,0,0,0,,[],(None)
Palisade_Connector,text_Palisade_Connector_name,,1,0,,1,1,1,Citizen,Palisade_Connector,Palisade_ConnectorUpgradeEpoch1,7,0,10,0,0,0,0,0,0,0,,[],(None)
Palisade_Gate,text_Palisade_Gate_name,,1,0,,1,1,1,Citizen,Palisade_Gate,Palisade_GateUpgradeEpoch1,28,0,40,0,0,0,0,0,0,0,,[],(None)
Palisade_Gate_Upgrade,text_Palisade_Gate_Upgrade_name,,1,0,,1,1,1,Citizen,Palisade_Gate_Upgrade,0,0,0,0,0,0,0,0,0,0,,[],(None)
Road,text_Road1_Piece_name,,1,0,,1,0,1,Citizen,Road,RoadUpgradeEpoch1,3,0,0,1,0,0,0,0,0,0,0,,[],(None)
Road_Flow,text_Road1_Flow_name,,1,0,,1,1,1,Citizen,Road_Flow,Road_FlowUpgradeEpoch1,3,0,0,1,0,0,0,0,0,0,0,,[],(None)
```



Save changes to the file dbtechreenode.csv

We open the game in the XII era. Let's see what can be produced in the barracks:



If you want, you can change the icon, give a new name and description for this unit. To prohibit the production of this unit in the XIII era, open the file EE2X\_db.zip\EE2X\_db\TechTree\epoch13\_upgrades.ddf

Insert the listed blocks where you want:

```
UpgradeObsoleteTech ObsoletePanzerGrenadier {  
    tech = PanzerGrenadier  
}
```

and

```
ObsoletePanzerGrenadier
```

to a large block

```
UpgradeRefSet MainEpoch13 {  
    upgrades = [ .... ]  
}
```

To learn more, read the following chapters.



### 3 What can be changed in the UnitType block

Suppose you already know that each unit in EE2 is defined by two consecutive text blocks: its **UnitType** and **UnitModel**. In short, UnitType is responsible for the information about the unit, and the UnitModel for its graphical appearance.

Consider **UnitType**  
Open any unit text block for clarity. It may contain the following single lines:

**parent** – parent, indicates the unit (another UnitType block) that is the basis of this unit. If some features are not registered in the new entity, they are taken from the parent entity.

**Example:**

parent = LeaderMilitary

**Values:** any, always points to a different UnitType block, it does not have to match an actual unit in the game.

**ownerType** owner type determines whether the unit belongs to players, to nature, or to common affiliation. In particular, this parameter affects the tab where the unit will be located in the map editor.

**Example:**

ownerType = PlayerOnly

**Values:** it can only take 3 values

ownerType = PlayerOnly - can only be owned by the player

OwnerType = WorldOnly - it can only belong to nature

ownerType = PlayerOrWorld - here and there

**placementType** the type of location determines where that unit (or building) can be placed, e.g. on land, sea, seashore, etc.

**Example:**

placementType = Dock

**Values:** various

placementType = Unplaceable - no placement, in fact the unit does not appear in the map editor. Heroes, dead animals, etc. have this parameter.

placementType = Dock - seaside accommodation

placementType = Building - arrangement by building type

placementType = DeepWater - placement under water, have combat units, fish

PlacementType = Bridge - in fact bridge type accommodation on the river bank.

PlacementType = NearWater - location near water, have naval defense

placementType = Wall - wall placement such as towers and gates

placementType = Farm - accommodation by farm type

placementType = Road - accommodation as a road

placementType = OnResource - placement on a resource. It is interesting that it allows you to put a building in place of any raw material and building. There is a line for this parameter where the resource is registered (can be any building)

placementTargetType = ResourceOil

**icon** - specifies the icon. This parameter is usually the last one. Specifies the name of the icon.

**Example:**

```
icon = icon_unit_SiegeGun
```

The icon file itself and its parameters are defined in two text files: `dbsprite_unitIcons_packed.csv` `EE2X_db.zip\EE2X_db\UI\PackedSprites` and `dbsprite_unitIcons.csv` `EE2X_db.zip\EE2X_db\UI\UnpackedSprites`

Identical lines are written in them

```
icon_unit_F16,icon_unit_F16.tga,0,0,60,60,0
```

Here 60\*60 indicates the size of the icon in pixels, `icon_unit_F16.tga` its actual icon file.

**Values:** various

**fullMapOverlay** defines sprites for some units (how the unit is displayed on the minimap)

**Example:**

```
fullMapOverlay { sprite = spr_fullmap_plane; scale = 1.5; }
```

**Values:** various

Now let's move on to the big block called **properties**

**properties** – it is a property block, there is a lot of different information stored here.

**Example:**

```
properties {  
    displayName = tx_utn_LeadTedRoos_name  
    displayNameScenEdit = tx_utn_LeadTedRoos_sname  
    hitpoints = 3000  
    verboseTooltip = vtt_unit_LeadTedRoos  
}
```

**Values:** consists of different sub-blocks (lines) that also have their own parameters.

Sub-blocks that define various text information for an entity:

```
displayName = tx_utn_heavyArt10_name
```

unit name in game, unit name is written in the manual

```
dbtext_unittypenames EE2X_db.zip\EE2X_db\Text
```

```
displayNameScenEdit = tx_utn_cmpgnAmerFieldGun_sname
```

unit name in the map editor

```
verboseTooltip = vtt_unit_siegegun
```

text prompts for the unit, recorded in the manual

```
dbtext_unittypetips EE2X_db.zip\EE2X_db\Text
```

Other sub-blocks:

**rps** role in the game

All existing rps categories with descriptions, see below.

**SizeX** = 3.3

**SizeY** = 1.0

the size of the circle actually determines the area occupied by the unit in the game, this area determines the amount of its interaction with the game world, in particular damage from explosive projectiles, the ability to pass through narrow passages.

**mass** a parameter that specifies the repulsion (not traversal) of one unit from another. If the value is 0, a unit can pass another person's unit through its circle.

**RallyPlacementFlags** = ( Terrain | Water | Mountain | AnyBuilding | AnyMobile | Allied | Neutral | Hostile ) – different tags for an entity.

**popCount** determines how many people (population) a given unit occupies. In addition to positive values, it can take 0 and negative values. It is not recommended to set negative values as it freezes the game.

**hitpoints** = Parameter that determines the value of health (life, strength).

**moveType** type of move.

It can take the following values:

moveType = Air such as planes and helicopters

moveType = kUnitMoveType\_Wheeled, e.g. artillery, robots

moveType = FootCitizen, for example a citizen

moveType = Mechanized a mechanized unit

moveType = Wheeled another on wheels, for example a battering ram

moveType = Mounted some leaders

moveType = WheeledTrade mobile commerce

moveType = kUnitMoveType\_Foot medical leaders

moveType = kUnitMoveType\_NavalDeep maritime

moveType = FootSpy spy

**FootprintType** = Specifies the shape of the rectangle underneath a unit or building that appears when you select a unit or building.

Ellipse – ellipse (circle) frame shape for units

kFootprintType\_Rectangle rectangle, frame shape for buildings

**mmColor** = [224,224,224,255] (?)

**stance** = Defensive the initial attitude of the unit. May

be Aggressive i.e. Aggressive tactic and Defen-sive

that is, a defensive tactic

HoldPosition Holding position tactics

Cautious A careful tactic

HoldFire Passive tactic

**properties** finished

Now let's move on to the big block **sounds**.

**sounds** determines the sound for different actions of units. It is not the sound of missile flight or tank movement that is determined here, but the sound the unit makes in-game.

**Example:**

sndCmdMove = ["v\_leader\_mil\_02\_com01" "v\_leader\_mil\_02\_com02"]

v\_leader\_mil\_02\_com01 the name of the sound, its characteristics and the sound file are saved in: dbsound.csv EE2X\_db.zip\EE2X\_db\Audio

Sample line from dbsound.csv:  
v\_NeAdvance01,NeAdvance01.mp3,0.75,0,0,1,0  
NeAdvance01.mp3 sound file mp3, 75 - sound volume.

**Values:** various

**sndCmdMove** Occurs after ordering a unit to move to a point.

**sndCmdAttack** Occurs after an attack order has been issued.

**sndSelection** Occurs when selecting a unit.

**sounds** finished

Another big block **abilities**

**abilities** it determines the abilities of unit, there are many different abilities

**Values:** consists of different sub-blocks (lines) that also have their own parameters.

**LOS** { range = 5 } range of view, the numerical value is indicated. However, there is a restriction for the player - the maximum view range is 15, the fog of war does not disappear beyond 15 distance in the game. At the same time, there is no such limitation for the computer player as well as for your units. So this limitation is purely visual.

Sub-blocks that determine the speed of different types of units:

**AircraftMove** { speed = 1.725; angSpeed = 180 } speed and speed of rotation (180 degrees per second) of the air unit.

**NavalMove** { speed = 1.725; angSpeed = 180 } speed and speed of rotation (180 degrees per second) of a naval unit.

**VehicleMove** { speed = 1.35; angSpeed = 180; followTerrainSlope = 1; bounce = 0.2; } the speed and rotation speed (180 degrees per second) of a land unit.

**Move** { mustMoveToTurn = 0; speed = 0.92; angSpeed = 180 } speed and speed of turning (180 degrees per second) of a land unit belonging to nature, such as a cow.

Garrison sub-blocks:

**Garrisonable** { garrisonTypes = [ AirportOrAircraftCarrier ] } The unit can enter buildings (it is a garrison). The type of these buildings is given in parentheses.

```
GarrisonAircraft {  
    numOfSlots = 4;  
    garrisonType = AirportOrAircraftCarrier;  
    GarrisonFlagModelName = "GarrisonShip"  
}
```

The building can accommodate other units, numOfSlots number of units accommodated, Airport or AircraftCarrier units.

Except **GarrisonAircraft** there are other types of garrisons.

Special effects and forces sub-blocks.

```
AreaEffect { effects = [  
    ]  
}
```

```
SpecialPower { effects = [  
    ]  
}
```

**AreaEffect** - effects (passive effect), the unit constantly has this effect. This effect usually works within a given radius.



**SpecialPower** - special forces (active effect), the unit can use this effect (press the special button)

All existing effects with descriptions, see below.

### **ConvertAttack**

ConvertAttack { convertTimeModifier = 3.0; range = 2.0; reloadTime = 4.0; applyDamageTime = 1.0; capture = 1 }

effect of capturing buildings available to infantry, also effect of unit conversion by priests.

**Resource** { type = kResourceType\_Food; alwaysExhaustible = 1; amount = 10; maxNumOfWorkers = 2 }

A subblock that defines that this unit is a resource. Only food can be a movable resource.  
type - type of resource

alwaysExhaustible - parameter defining an infinite resource or not. 1 means the resource is finite, 0 means infinite.

amount - amount of resources.

maxNumOfWorkers - the number of people who can extract a given resource at the same time.

**Abilities** finished

Another big block **attributes**

**Attributes** – attribute block, actually another block of unit abilities.

**Example:**

```
attributes [  
    GoesOnAirMissions  
]
```

**Values:**

GoesOnAirMissions plane return mission to the airport

CanJoinHeroArmy the unit can become a hero

HealedByMedic the unit can be healed by a medic

RectangularFootprint the unit has a rectangular frame when selected (?)

HealedByHERC the unit can be healed by a robot.

IsWonder is a wonder.

**Attributes** finished

**UnitType** finished

## 3.1 RPS List

All existing rps categories are described.

//	Null	without rps
//	LightInfantry	light infantry
//	HeavyInfantry	heavy infantry
//	LightArtillery	light artillery
//	AntiAircraft	anti-aircraft unit
//	HeavyArtillery	heavy artillery
//	LightMounted	light mounted units
//	HeavyMounted	heavy mounted units
//	Citizen	workers
//	Building	buildings
//	LandNonCombat	non-combat land units
//	Animal	animals
//	CloseAirSupport	air support
//	AirSuperiority	air superiority
//	Bomber	bomber
//	AtomicBomber	atomic bomber
//	AirNonCombat	aviation not combat
//	Submarine	submarines
//	SubmarineNuclear	nuclear submarines
//	WarGalley	war galleys
//	FightingSail	combat sailing ships
//	Frigate	frigates
//	Galleon	galleons
//	Battleship	battleships
//	Destroyer	destroyers
//	Carrier	aircraft carrier
//	Transport	transporter
//	SeaNonCombat	sea non combat
//	SpecialForces	special forces
//	Ram	ram
//	BuildingCombat	combat buildings
//	Helicopter	helicopters

## 3.2 List of effects

All effects (passive effects) and special effects are described here. abilities (active effects) directly related to a unit (or building). The global effects of the crown, flags, regions, territories, weather are not taken into account. All considered effects can be used in modding. There are two types of abilities: active (forced by pressing a button) and passive (constantly active).

All effects are broken down into effects:

- heroes
- leaders
- scenario
- units
- wonders
- other effects

All numerical parameters described for the effects can be changed at your discretion.

### Wonder effects

#### WonderPowerAfrican\_1

Manyatta

Every minute a new light infantry fighter emerges from this structure. With a probability of 20%, it can be replaced by a representative of a unique type of cultural troops who owns the structure.

What can be changed:

You can change the probability 20% of the chance depends on 4 light infantry units + 1 unique unit.

You can remove the production ban with maximum population.

New unit production time.

#### WonderPowerAfrican\_6

Market in Jenna

The owner of this building receives one amount of gold for each transaction made by its merchants in all markets of the world.

What can be changed:

the default resource type is gold (you can change the resource type).

The default amount of gold received is 1 (the amount of resource received can be change).

#### SpoilsOfWar

Brandenburg Gate

War trophies

By destroying any enemy unit or building, you receive 20% of its cost in technology points. Duration - 60 sec.

What can be changed:

in fact it says: 1 technology point for 5 killed units, so it's 0.2 technology point per killed unit  
For example, you can change a resource from tech to gold.  
You can change the amount of resources received for killing another unit, by default it is 0.2.

### HomelandDefense

#### PENTAGON

Houses have the ability to shoot the enemy units. Nothing can be changed.

### RiddleOfTheSphinx

#### SPHINX

All buildings are immune to the capture effect.  
Nothing can be changed.

### SultansReward

#### Selemi Mosque

When an enemy unit \ Building is destroyed, you get 20% of its construction cost.

What can be changed:

resource = Gold you can change the resulting resource

bonus = 0.0 percentage optional by default 0

scale = 0.2 percentage of the cost

### LifeToTheDesert

#### Desalination station

Citizens gather food 50% faster and the maximum population increases by 35. So it introduces 2 effects:

1 increase in the maximum size of the population (but not more than the game parameters allow)  
by default 35 - editable

2 more efficient food collection

the type of resource collected is editable

Percentage of the yield of the resource collected (can be less than 50%)

### Farseeing

#### Moon and stars tower

#### Omniscience

Increases the view radius of all buildings.

The ability to see everything that is happening around the town center of other players.

What can be changed:

you can set the duration of the effect - it's strange for a wonder (?)

It consists of two minor effects.

The default view radius is 14, but you can change it.

### Splendor

#### The Forbidden Palace

No resource loss when one of your buildings is captured.

By occupying another player's building, you get an additional 35% of the seized resources.

What can be changed:

You can set resource loss as the default percentage 0

You can set the percentage of resource seized resources, the default is 35%

### AutomationEpoch11

#### ROBOTICS FACTORY

Every minute, a randomly generated unit is created in the Robotics Factory. What can be changed:  
delay = 5 how much time to produce the next unit.

unitTypes = [ "LightInfantryHero" ] type of unit produced.

### AutomationEpoch15

#### ROBOTICS FACTORY

Every minute, a randomly generated unit is created in the Robotics Factory.

unitTypes = [ "Tank" "HeavyTank" "HeavyMounted3" "LightMounted3" ]; - several different units can be produced.

### ArtOfWar

Temple of the great jaguar

All your attacking troops have the same morale bonus as defending troops. Nothing can be changed.

### Proselytizing

Great Pyramid of Tenochitlan

Whenever an enemy unit is re-captured, there is a 5% chance that any unit within X radius of the newly converted unit will also be reflected immediately.

You can change the conversion radius and 5% chance. You can put availability not all over the world - but separately for a person.

### EnergyEfficiency

Solar station

All construction costs, repair costs reduced by 50%.

It's changeable 50%

Other wonder effects

### RapidDeployment

Rapid deployment

All troops stationed in the Pentagon can be immediately transferred to any other location on the same continent or island.

costs 2 seconds.

### HardTeachings

Provides 0.1 points for killing an enemy unit. 10%

Can be matched.

### Unit effects

### HERC\_AntiMissileSystem

Electronic Warfare Herc has a built-in missile defense system, standard robot capability.

Active power



What can be changed:

default shield radius 4

The shield can be assigned an amount of hit points, but the default is 0, which means it can withstand unlimited damage.

duration of the effect

### HERC\_Shield

Herc repair robot can install this shield.

What can be changed:

default radius 2

Absorbs attacks 1000 dmg (adjustable) before disappearing.

Duration of the effect

Active power

### HERC\_EMP\_blast

Electromagnetic pulse

Running time 15 (adjustable)

default radius 5

It consists of many small effects

Active power

### Nuclear\_fallout

nuclear explosion, radioactive contamination zone effect.

Radioactive fallout near a spawning player (born or just spawned). A zone of infection is created that affects all your and enemy units for 30 seconds (time is adjustable) in a specific area, damage per second is adjusted (default 15), default radius is adjusted on 6. Precipitation stays in place and does not follow unit who created him.

The unit is also vulnerable to precipitation. Does not affect buildings. War against neutrals is not declared after contamination.

Duration 30 sec

It consists of many small effects:

impact on the unit itself, radius 6 (why?) damage per second 15

impact on other units within the radius 6 damage per second 5

### Nuclear\_fallout\_nonhuman

This effect does not infect other units. The unit is contaminated with radioactive fallout for 30 seconds at 5 damage per second. The unit does not infect anyone, it just dies.

Duration 30 sec

damage per second 5

### GatherIntelligence

Espionage

Gathering information in an enemy building. If successful, all buildings of that type belonging to the same enemy will become visible. In addition, if espionage was committed in the town hall, you can receive information about the enemy's ratings and inventions within 5 minutes.

Active power

the time it takes to gather information is 5 sec

Duration 300 seconds

### ToxicSpill

#### Infection

Infection of hostile warehouses and town centers with a dangerous virus. Workers on the grounds the infected get sick and lose their health. Effect used by spies as a special power, in the case of a passive effect - a yellow circle under the unit, the unit does not die by itself, but infects its citizens who die (if the initially infected unit is a worker, the same thing dies), does not infect foreigners, infection lasts 40 seconds (time is adjustable) infected workers do not infect others.

Active power

Duration 40 sec

radius of infection 4

time of infection 5 sec

### ToxicSickness

#### Disease

The combat unit is infected with the virus and loses health. It cannot infect anyone, it only dies by itself, duration 20 sec.

radius = 2 radius of infection of an already infected unit.

DamageAmount = 7 damage dealt to a unit over 1 second.

### ImprovedToxicSpill

#### Improved Infection

Infection of hostile warehouses and town centers with a dangerous virus. Workers in contaminated areas become sick and lose their health.

Is similar to ToxicSpill, difference: you can adjust the infection time of other players, the default is 5 seconds

duration 40 sec

radius = 4 the radius of infection is here.

### ImprovedToxicSickness

Is similar to ToxicSickness does not seem to be different.

duration 20 sec

radius = 2

DamageAmount = 8,4 damage dealt to a unit over 1 second.

### ToxicCleanup

#### Sanitization

Disinfection of warehouses and town centers.

Active power

disinfection time is 5 seconds

### SabotageBuilding

#### Sabotage

Sabotage in the enemy building. A building that has suffered from sabotage is temporarily suspending its work.

Active power

An effect used by spies as a special power in the event of an effect applied to units (that is, if the effect is written in passive), the unit cannot do anything but move. For example, it cannot attack with the alt and mouse keys, the building action buttons are not active - gray.

maskSelection = Buildings The parameter can be changed, for example, you can place ground units LandRPS, and then the spy can stop not only buildings, but also ground units.  
Action time 120 sec  
sabotage time 5 sec

### ImprovedSabotageBuilding

Improved Sabotage

Sabotage in the enemy building. A building that has suffered from sabotage is temporarily suspending its work.

Active power

The duration is 144 seconds, should be 20% longer than the normal sabotage time.

sabotage time 5 sec

### SabotageBuildingUnique

Unique Sabotage

Sabotage in the enemy building. A building that has suffered from sabotage is temporarily suspending its work.

similar to SabotageBuilding used for Mi-6

duration 180 sec

sabotage time 2 sec

### RestoreBuilding

Restoring buildings

Restoring the functioning of buildings damaged by enemy sabotage.

cooldown 2 seconds

Active power

### Indestructability

Indestructibility. The unit cannot be damaged, it cannot be destroyed, but you can turn it to your side (catch, convert).

However, in the parameters you can define the damage caused to the unit. The default is 0. Oddly, it was originally an active power (?)

### HealGroup

General treatment

The target unit adds 50 health (life) to all friendly units within 3 hexes of it. There must be an Aztec shaman in this area. A shaman is not required.

Active power

radius 3

hitpointsHealed = 50 – number of health units restored

has its own button

### BreathOfDragon

Dragon's breath

2x more damage dealt by the unit when shooting.

action time 30 seconds

It only works as a passive force

### DeadlyAim

## Steady hand

Unit damage doubled. It only works as a passive force  
duration 30 sec  
scale = 2 - twice as high

## Forage

### Forage

By gaining food, a combat unit restores health.  
duration 30 sec  
But for some reason, this effect doesn't work.

## BlessPreviewGlow

effect used by priests as a special power in the case of an area effect, the location where the unit first spawned has the ability to give blessing circles to other units approaching that location. In fact, it is only a visual effect.  
radius of action 2  
luminescence frequency 0.2

## Bless

### Blessing

All your units within the radius of this effect deal more damage to the enemy. An effect used by priests as a special power, in the case of an active effect, the unit simply has a blessing.  
Active power  
radius of action 2  
duration 20 sec  
time to get the effect 4 seconds

## BlessUnits

like Bless, nothing seems different  
The default size of the bonus is specified here 1.2 (20% attack bonus)

## Extraction

### Evacuation

Your agent is trying to navigate to the nearest evacuation point.  
If successful, it will immediately go to your main town center. Active power  
Duration 15 sec  
distance from the unit where the evacuation point is located 10 The size of the evacuation zone 0.6

## CityHeal

### Native walls

Gradually restores health (life) to all friendly land forces in the city.  
radius = 0 apparently defined elsewhere

## CityResistCapture

### Confrontation

Increased protection of all buildings against enemy capture

radius = 0    apparently defined elsewhere

### SimpleGlow

Blue circle effect, the circle can be changed

### KingRegicideGlow

Transparent cup effect as in Bless

## Script Effects

### Maasai\_Tough\_Walls

Masai hard walls

there are two values, both 0,1

Reduces the damage dealt by the enemy in fractions 0,1 (10 times less)

### Maasai\_Herd\_Heal

Deals damage to all soldiers in the specified territory (a certain amount of damage per second).

duration 10 sec

radius 20

Damage per second 5

**HighFervorPlayer** - the effect is equivalent to the high temperature of the leader.

### OasisHealing

Health regeneration for all ground units within a radius 15

restores 3 life points per second

### RussianWinter

The effect of the Russian winter of freezing French troops during the winter. The French army was not ready for the harsh Russian winter. Now she is helpless.

duration 360 sec

radius 5

Creates red circles from the enemy across the whole map.

### CitizenCapacityIncrease

Doubling the maximum amount of resources carried by an employee. increases the number of employees from 15 to 30

What can be changed:

increase in how many times you can change

### Invincibility

Inviolability, The unit becomes inviolable

Although the value can be changed, the default is 0

### InvincibilityAnyPlayer

Similar effect, but only similar to other players (?)



### DamageUnitOverTime

Loses life over time

damage taken per second 1.

### DamageUnitsOverTimeNearLocation

Damaging all troops in the specified territory (specific number of damage units per second).  
duration 10 sec

### ChrisBsHealAreaEffect

This is a healing effect.

It is only intended to heal a few units after the battle.

duration 10 sec

radius 15

health per second 1

### MarienburgHealing

Heals all units at a distance, you can change the distance. the default distance is 40.

3 health per second

### American8\_ImmuneToNuke

No damage from nuclear contamination, although the damage parameter can be adjusted.

### PreventDeletion

Immune leaders! (?) Nothing

can be changed

### RefugeeSpeedModifier

Reduces the speed of the unit many times. There is a value of 0.5 - it means a double decrease.

### SupplyCartSpeedModifier

Reduces the speed of the unit many times.

Value 0.6

### ImmuneToCapture

Building immunity.

Nothing can be changed.

Scenario Effects Completed.

Various Effects. Other Effects

### ResistanceGlow

Resistance animation effect - Increasing capture protection - For all units (even other states) blue circles under the unit.

Nothing can be changed

### CaptureGlow

Resistance animation effect for captured buildings.  
animation of capturing the ring of fire building under the unit.  
Nothing can be changed

### TradeWaitGlow

Animation effect of the trade objects while waiting for their turn to trade. Nothing can be changed

Misc effects completed

### Leader effects

### Mobilization

Mobilization  
Significantly increases the speed of training troops in a specific building.  
Active force.  
Active power  
Production time reduction by 0,5 (by half)

### ForeignInvestment

Foreign investment  
Ability to instantly purchase enemy or neutral buildings (excluding town centers, fortresses, outposts, and large structures)  
for double cost.  
Active power  
You can change the cost

### Capitalism

Capitalism  
Ability to instantly outbid an enemy or neutral combat unit for double the gold cost.  
Active power  
You can change the cost

### Endeavor

Endeavor  
Complete half of the construction work for a specific town center or large structure immediately.  
Active power  
You can adjust the percentage of immediately completed tasks by 0.5 to half

### FastTracking

Fast Tracking  
Immediate completion of the construction of a specific object (except for town centers and large structures).  
Active power  
You can adjust the percentage of immediate job completion 1,0 - 100%

### Imperialism

Imperialism  
Indestructibility and 100% protection against the capture of the indicated friendly building.

Active power  
duration 45 sec  
The percentage of the vulnerability can be adjusted

### Rally

First aid  
Instantly restores 20% health (health points) to all friendly ground troops near the hero.  
Active power  
radius 3

### FireAndManeuver

Logistics  
Increases the movement speed of all friendly ground troops near the hero and prevents the movement of enemy ground troops.  
Active power  
duration 45 sec  
speed increase 1,5  
radius 5

### Diversion

Disinformation  
Create a fake squad of soldiers in the place you choose. This unit can be commanded in the same way as the real one.  
Active power  
duration 45 sec  
formation of phalanx units  
4 types of units can appear. by 11, 12, 13, 14-15 eras.  
number of new units.

### Customs

Trade Inspection  
Your markets adjacent to a hero bring double the income from trading.  
radius 5  
Generated Income - Gold  
income - 2

### Prioritization

Shock construction  
Improves the efficiency of construction and repair of all objects near which the hero is.  
radius 5  
increase 2 times

### MANAGEMENT

Organization of work  
Increasing the efficiency of extraction of all types of raw materials, in addition to information in area around the hero.  
radius 5  
increase 2 times

## Loyalty

### Loyalty

Your troops next to the hero cannot be converted, intercepted by enemy clergymen.  
radius 5

## Lectures

### Lectures

Improving the effectiveness of scientific work in temples and universities next to the hero.  
radius 5  
increase 2 times

## Nationalism

### Nationalism

If this hero is in one of your territories, it is because of patriotism the morale of all your soldiers in the same territory will increase significantly.  
damage reduction by 0.8  
reduction of damage due to fracture by 0,8  
increased attack power by 1,2

## HighFervor

### High Fervor

Increased attack power of all friendly soldiers near the hero. Increase also the attack power of buildings.  
radius 3  
increased attack power 1.2

## HighMorale

### High fighting spirit

Friendly ground troops near the hero take less damage from direct enemy units hits.  
direct hit damage 0,8  
blast damage 0,8

## Discipline

### Discipline

Friendly ground troops near the hero take less damage from enemy hits in the area.  
radius 5  
blast damage 0.5

## Resourcefulness

Inspiration (AoE) Doubles the regeneration rate of priests and other special units within 5 squares of the hero.  
radius 5  
loading time

## PropagandaPreviewGlow

Party color change

Effect to change the color of converted soldiers. Enemy units have shiny circles  
radius 3  
dot 0,2 – (?)

### Propaganda

Propaganda

Instantly convert all enemy ground forces to your side to specified area within 3 squares. The more the enemy lags behind in development, the greater the likelihood of success. Using only the passive effect as an active effect will crash the game. After birth, he immediately relapses, but then does not work.

radius 3

probability of passing from another era 0.6

time of capture

### Resistance

Resistance. When trying to take over certain buildings, enemy infantry loses 50% health  
amount of health loss 0,5

### SARS

SARS (test)

Immediate death of all enemy units within 3 spaces of the hero. Neutrals don't die. It only kills citizens.

radius 3

citizens can be changed to other units.

### SuperSight

Eagle Eye. Doubling the view radius, then increasing it by 5. Only works for the unit itself. However, it will still not be more than 15.

### Comments:

In addition to the numerical values, other parameters can be adjusted. For example, the mobilization effect is accompanied by a visual effect - green crosses on all treated units. You've certainly seen this visual effect. But this effect can be changed to other visual effects. Here is a list of visual effects that can be used for modding.

List of FX effects. All of these effects are defined by BaseModel text blocks. To test them, simply replace the FX\_BeingHealed entry in the raid text block with one of the following and apply the in-game raid effect to any unit that is not at full health.

FX\_BloodSpurt1 fragment scattering effect

FX\_BloodSpurt2 scattering effect of other fragments

FX\_Conversion conversion effect, the unit is covered with a blue-green cloud

FX\_BeingConverted conversion effect, blue-green cloud descends on the unit

FX\_BeingCaptured green smoke screen effect over the unit, the effect is huge





FX\_BeingHealed healing effect, green crosses over unit.

FX\_GroupHeal two green crosses above the unit

FX\_Wheat\_Harvest\_1 grass mowing effect

FX\_GuidedMissile\_Explosion\_1 huge explosion effect



FX\_ExhaustSmoke little smoke behind the unit  
FX\_ExhaustSmoke1 little smoke behind the unit  
FX\_ExhaustSmoke2 little smoke behind the unit  
FX\_ExhaustSmoke3 little smoke behind the unit  
FX\_Smoke\_Chimney\_Small high white smoke





FX\_Smoke\_Chimney\_Large high black smoke

FX\_Campfire\_Flame a fiery flame emerging from the unit



FX\_TrailingSmoke little smoke behind the unit  
 FX\_Falling\_drooping leaves from a great height



FX\_Citadel\_Connector - the appearance of a huge green tower

FX\_Citadel\_Tower - the appearance of a huge green tower

FX\_Citadel\_Gate - the appearance of a huge gate

FX\_Tolerance\_Priest the unit is covered with a blue-green cloud with rainbow rings

FX\_Tolerance\_Target a blue-green cloud with elongated stripes descends on the unit

FX\_Scientific\_Freedom a huge blue-green ring of balls around the unit





FX\_Mercantilism nothing

FX\_Mercantilism\_tradecart a huge gold coin appears nearby

FX\_RapidDeploymentCrate the chest is parachuted from a low height

FX\_Overtime\_Airport some details emerge

FX\_Overtime\_Mill some details emerge

FX\_Overtime\_Dock some details emerge

FX\_Overtime\_Stable some details emerge

FX\_Breath\_of\_Dragon the sign of the dragon appears

FX\_Deadly\_Aim sign appears

FX\_Forage sign appears

FX\_Bless A blue ring appears above the unit

FX\_FiringCannon little effect of exploding projectile

SupremeLeaderGarrison\_Imperial a huge icon of the imperial crown rotates around the unit

SupremeLeaderGarrison\_Military similarly for this crown

SupremeLeaderGarrison\_Economic similarly for this crown

SupremeLeaderUnit\_Imperial similarly for this crown a small icon

SupremeLeaderUnit\_Military similarly for this crown a small icon

SupremeLeaderUnit\_Economic similarly for this crown a small icon

FX\_GuidedMissile\_Trail\_1 a fiery green beam reappears

FX\_ARROW\_FIRE\_Trail\_1 small explosions appear in front of the unit, like from a machine gun

ThePatrolPointMarker huge patrol badge

You can also change the button (for active effects), the mouse cursor, the icons in effect:

button = "unit\_power\_antimissile" active force button, the button parameters are set in

dbunitactions.csv



mouseCursor = HercShield mouse cursor  
Icon = icon\_power\_emp icon

List of all buttons available in the game for active forces:

unit\_power\_propaganda, this button is identical to bless  
unit\_power\_healGroup, new button, group treatment.  
unit\_power\_bless, bless button for priests  
unit\_power\_fireAndManeuver, button for the leader  
unit\_power\_rally, button for the leader

Spy buttons:

unit\_power\_gatherIntel, espionage  
unit\_power\_extraction, evacuation  
unit\_power\_decontaminate, disinfection  
unit\_power\_restore, return after sabotage  
unit\_power\_contaminate, infection  
unit\_power\_sabotage, sabotage  
unit\_power\_sabotageUnique, no button at all

Scoreboard button:

unit\_power\_endeavor,  
unit\_power\_fastTrack,  
unit\_power\_capitalism,  
unit\_power\_forinvest,  
unit\_power\_mobilize,  
unit\_power\_imperialism,  
unit\_power\_diversion,

unit\_power\_rapidDep, Fast fortress setup icon

Several other parameters that can be changed.

There is a maskSelection parameter that determines what is affected by this effect.

Possible value:

maskSelection = Buildings all buildings  
BuildingsThatProduce all production buildings  
LandRPS all ground units  
AllHercs all robots  
LandNonFootUnit all ground non-foot units

You can also change the circle under the unit. This circle is defined by the parameter GlowType.  
Its possible values are:

kGlowType\_Good, good - blue circle  
kGlowType\_Bad, bad, orange circle  
kGlowType\_Mixed, without a cup  
kGlowType\_Neutral, shiny white circle resembling a priest blessing effect  
kGlowType\_King, shiny powerful white circle similar to the priest blessing effect  
kGlowType\_HeroArmy, star to match the player's color  
kGlowType\_HeroArmyLeader, star to match the player's color  
kGlowType\_SupremeLeader, pentagon (regular pentagon) to match the player's color

## 4 What can be changed in the UnitModel block

Let's analyze the block **UnitModel**

It may contain the following single lines:

**parent** – parent, like UnitType, points to the unit (another UnitModel) that is the base for that unit. If some features are not registered in the new entity, they are taken from the parent entity.

**Example:**

Parent = HeavyInfantry1\_05

**Values:** any, always points to another UnitModel, it does not have to match an actual unit in the game.

**ScaleType** scale type or height type. Basically, it defines the height (size) of a given object as the size of another reference object.

**Example:**

ScaleType = kScaleUnit

**Values:**

kScaleIdentity, Scale = 1.0f if there is no parent, otherwise it is a parental scale

kScaleUnit, // standard unit size

kScaleLandAnimal, // scales like land animals (cows, llamas etc.)

kScaleBuilding, // building scale

kScaleWall, // the scale of the walls is separate from the buildings

kScaleVehicle, // scale of wheeled units

kScaleMountedUnit, // scale for mounted units

kScaleTree, // scale for trees

kScaleAmmo, // scale for bullets

kScaleHero, // scale for heroes

kScaleHeroVehicle, // scale for heroes on vehicles

kScaleHeroMountedUnit, // scale for mounted heroes

**DefaultModel** model file, pointing directly to the model file of unit.

**Example:**

DefaultModel = CIV1\_Samurai\_Japanese.nif

**Values:** various, all 3d models of the game in nif format are in the graphics archive C:\Program Files\Sierra\Empire Earth II\zips

### **UIRenderingParams**

User interface related parameter.

**Example:**

```
UIRenderingParams {  
    xOffset = 0.01  
    yOffset = -0.66  
    zOffset = -4.50  
    scale = 3.20  
}
```

Now let's move on to the big block **States**

**States** contains many visual features such as animations, sounds etc.

Includes **StateName**. StateName – it defines in fact a specific action of unit, for example, when a unit walks, runs, attacks, cuts wood, builds, etc.

Example:

```
{      StateName = Idle      AnimName = CIV1_Samurai_Japanese_idle01.KF
      AnimVariants = [ CIV1_Samurai_Japanese_idle02.KF
                      CIV1_Samurai_Japanese_idle03.KF ]
      AnimVariantsWeights = [ 0.2 0.2 ]
}
```

**Values:** StateName can take the following values (not all are listed):

Idle - idle animation of calm or idle (unit is at rest, just standing).

Walk - movement accompanied by various sounds, action time

Death - unit death

Run - launching the action

Flinch - shudder, possibly from the impact of an exploding bomb

CaptureAttack1 - animation for capturing e.g. buildings

Attack1 - unit attack

Pack - packing animation (e.g. arquebusier)

Unpack - animation of unpacking, unfolding

TakeOff - plane take off animation

State structure

**AnimName** the name of the animation file,

**AnimVariants** different animation options for the same action

**AnimVariantsWeights** frequency of appearance of animation variants relative to the first variant.

**TextKeys** – animation accompaniment, here, for example, sounds are indicated

Example:

```
TextKeys = [ { Name = PLAY_SOUND Data = fx_sword_swing KeyTime = 0.40 } ]
```

Name - outgoing type

PLAY\_SOUND - the sound is played

Data - indicates some data, for example the name of a sound

Data = fx\_sword\_swing

KeyTime – audio delay time

**ChildNames** Visual effects are indicated

Example:

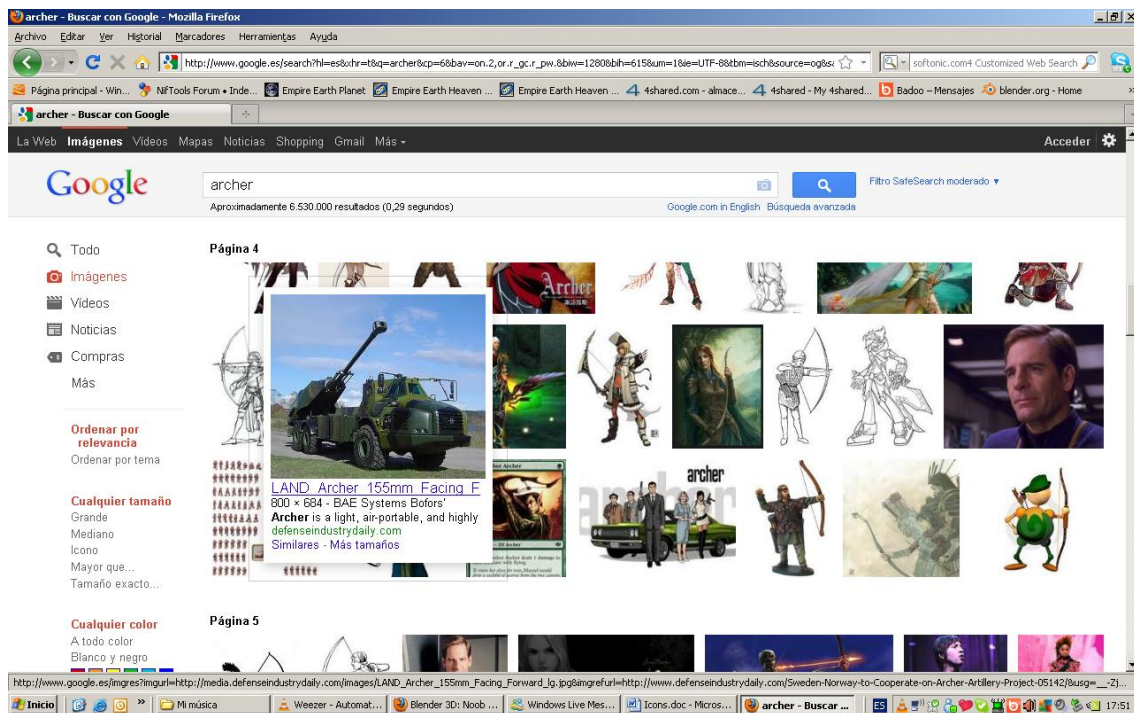
MachineGunTracer Tracking lines from a machine gun

For the most diverse units or buildings, you can specify the effects of other units.

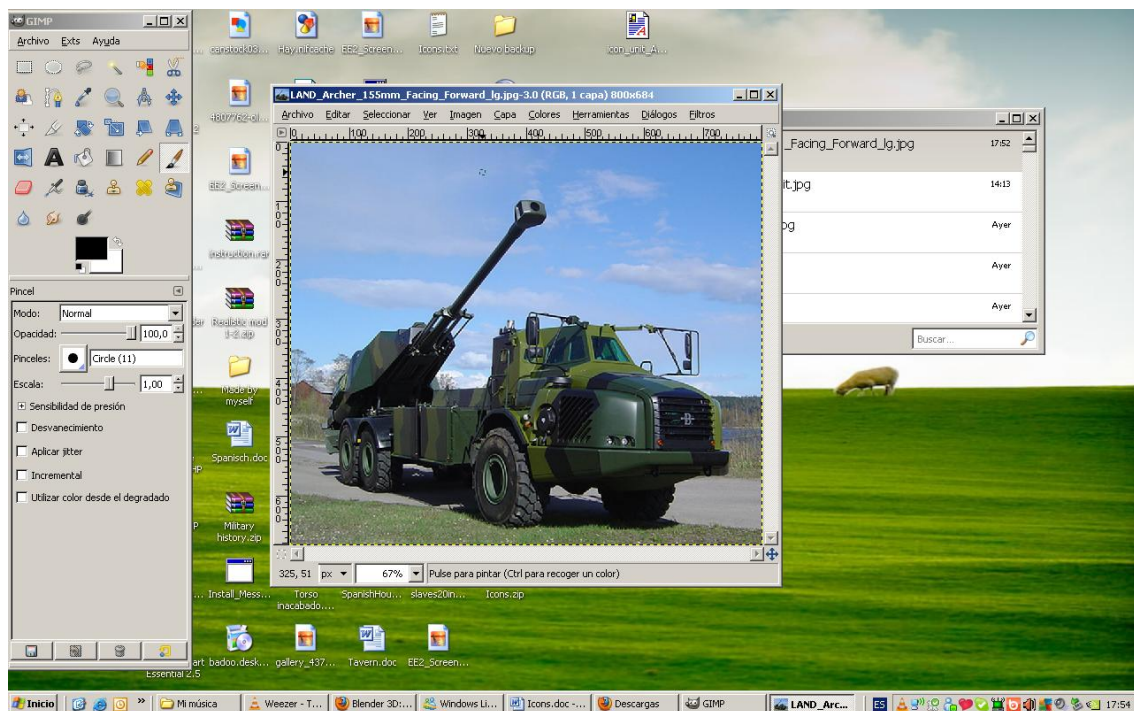
## 5 How to add a new icon to the game.

Author Michael134

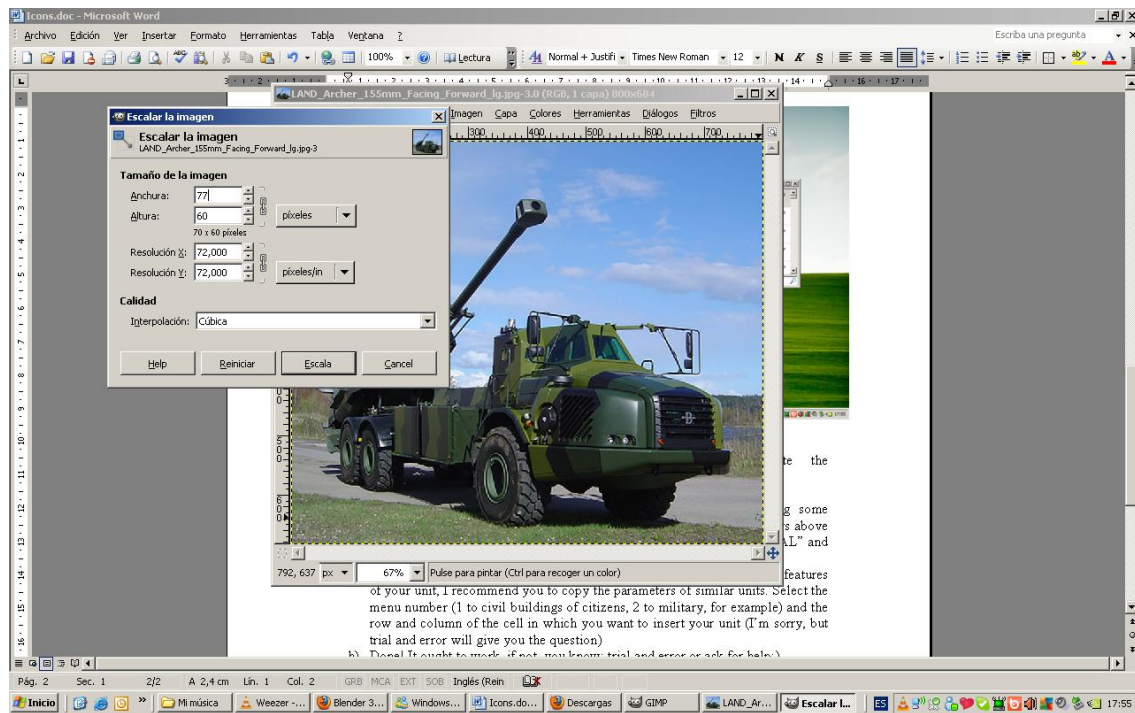
Let's find on the Internet a picture suitable for the icon of our unit



Resize the image in Gimp, Photoshop, or any other image editor to x = 64 pixels, y = 64 pixels.





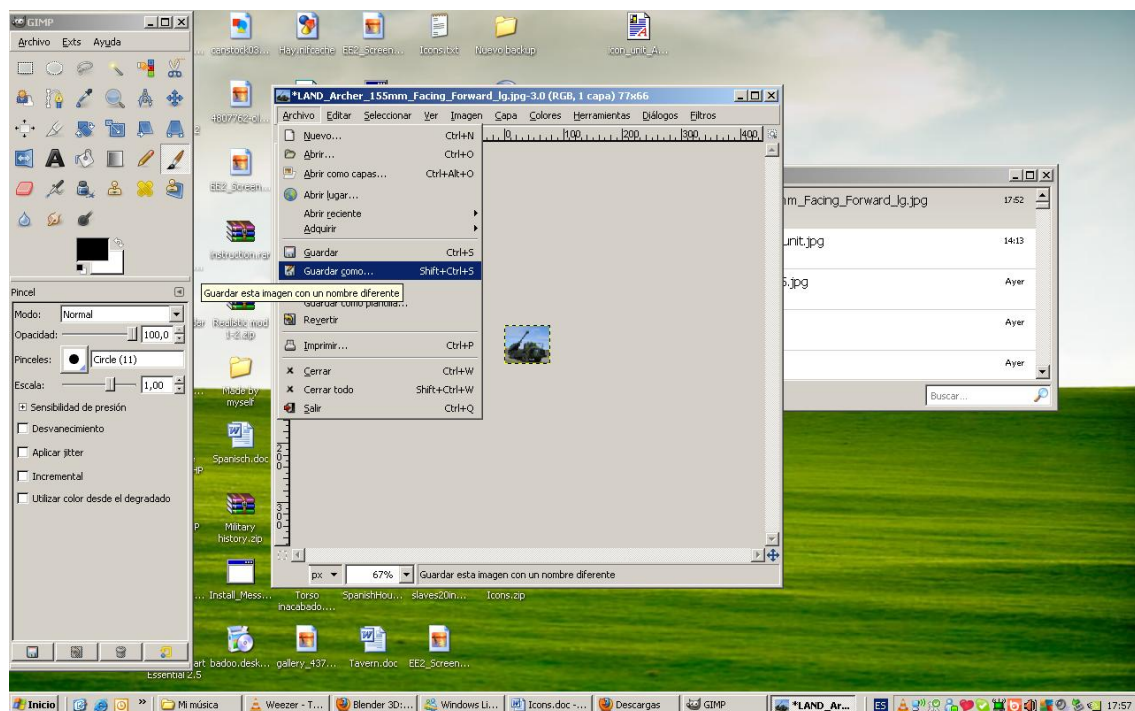


Save it in tga format, for example "icon\_unit\_yourunitname.tga". The game supports icons in TGA format

Place it in the Zips folder located at:

C:\Program Files\Sierra\Empire Earth II\zips

Place in file Textures.zip.



Open file dbsprite\_unitIcons\_packed.csv

Located in:

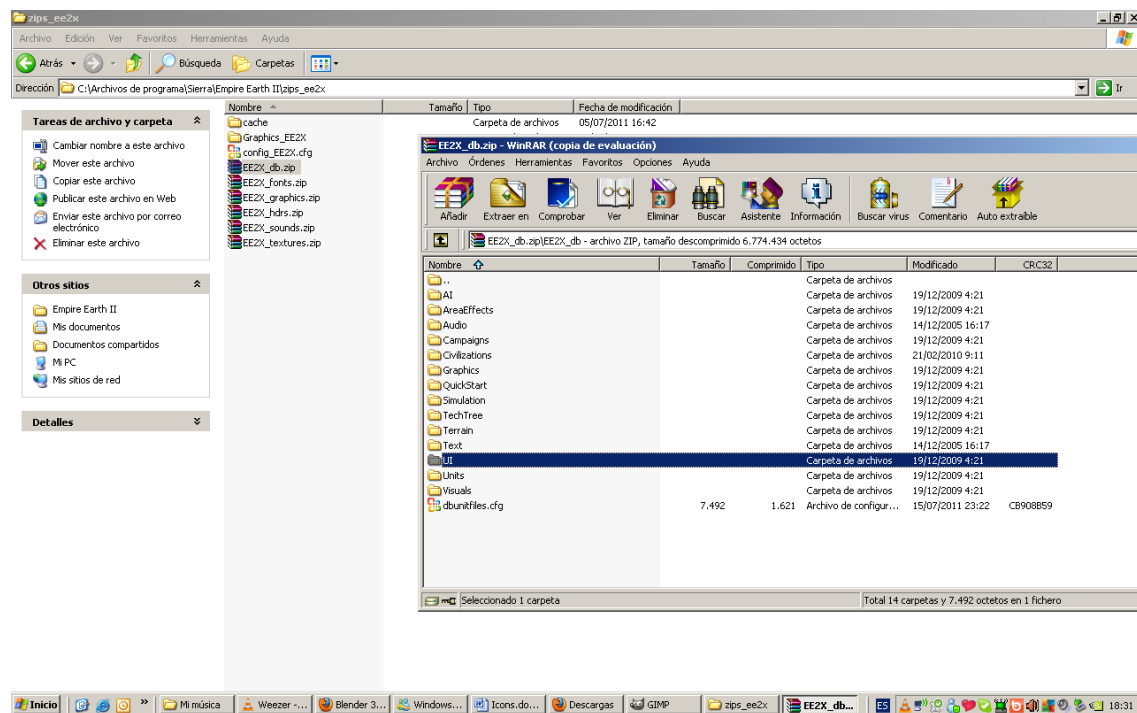
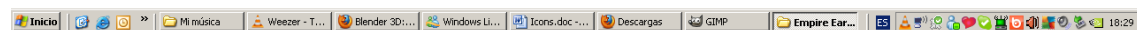
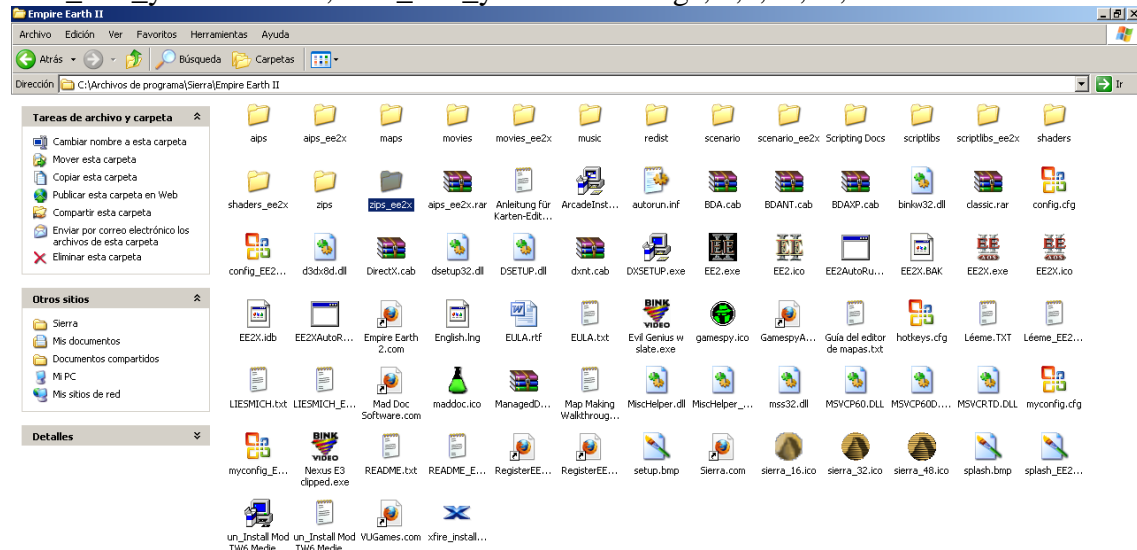
C:\Program Files\Sierra\Empire Earth II\zips\_ee2x\EE2X\_db.zip

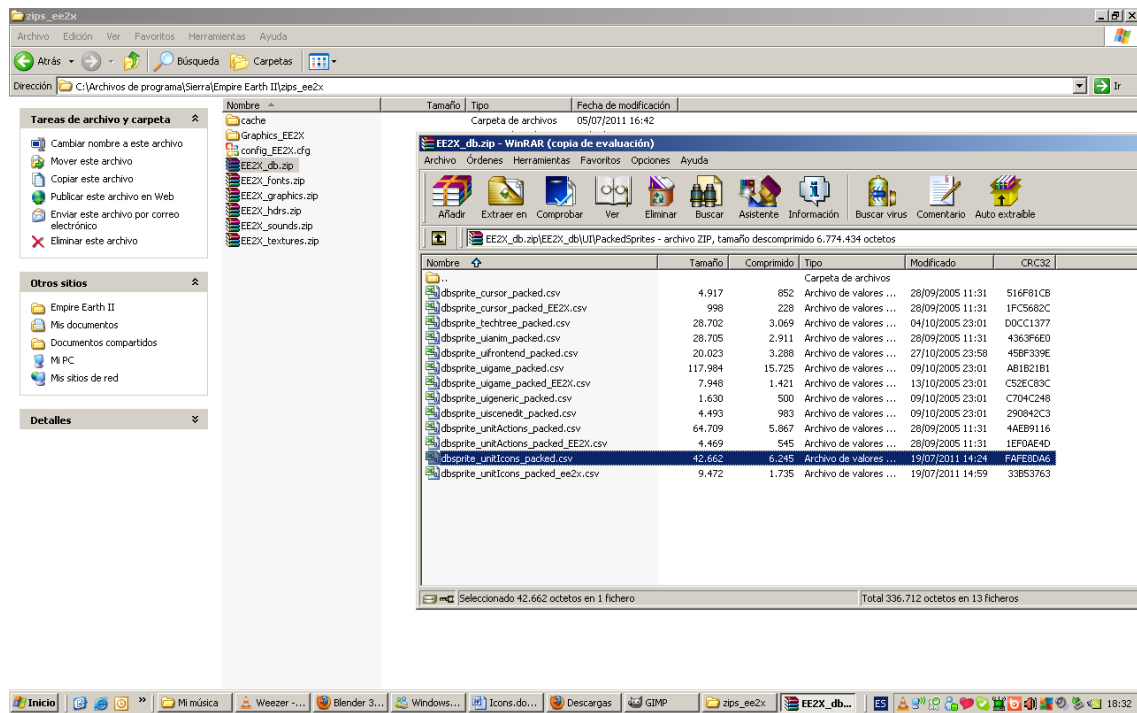
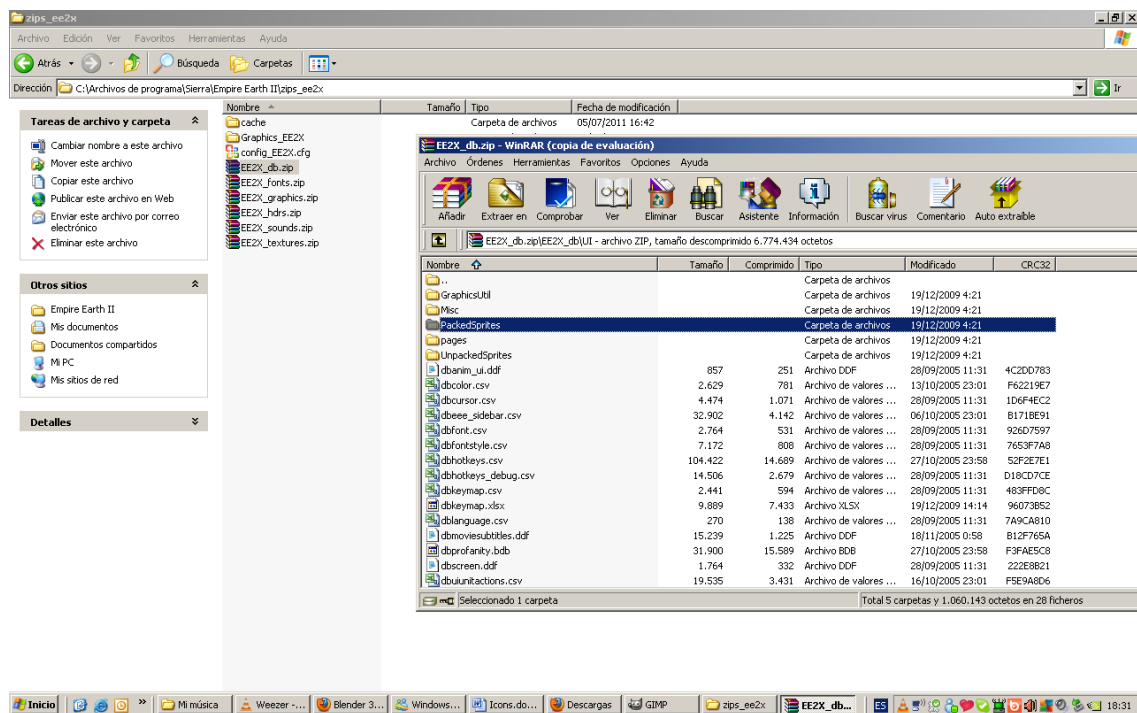
And next:

EE2X\_db.zip\UI\PackedSprites\dbsprite\_unitIcons\_packed.csv

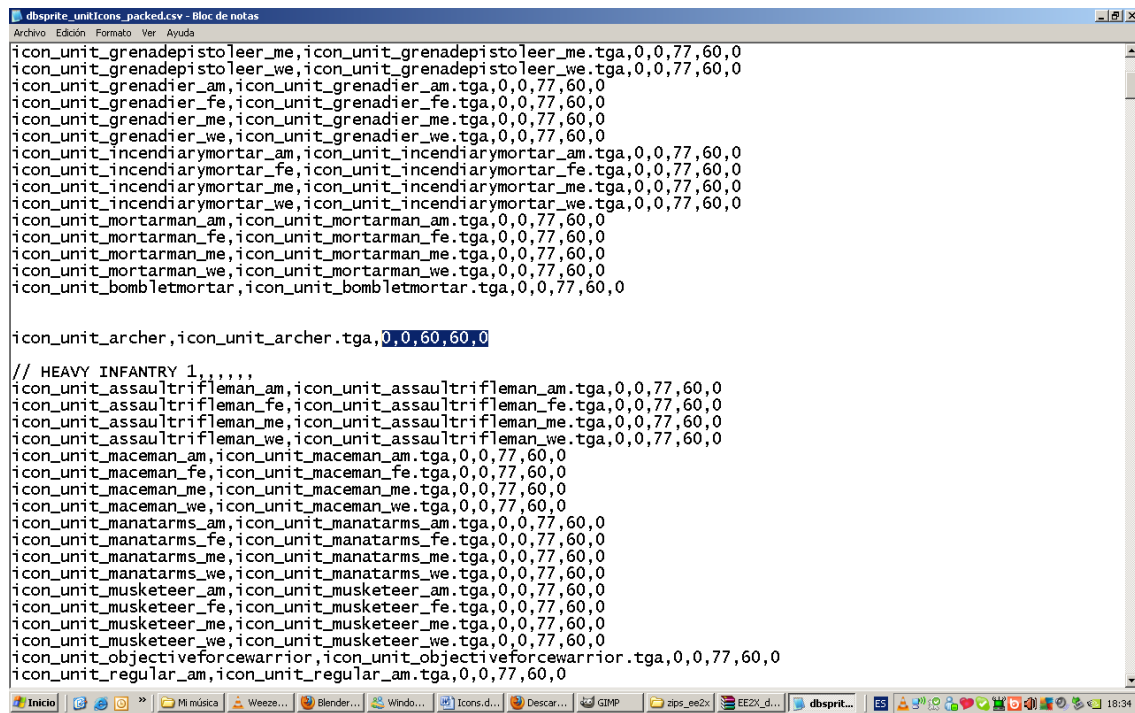
And put the following line in it:

icon\_unit\_yourunitname, icon\_unit\_yourunitname.tga, 0,0,60,60,0









```
icon_unit_grenadepistoleer_me,icon_unit_grenadepistoleer_me.tga,0,0,77,60,0
icon_unit_grenadepistoleer_we,icon_unit_grenadepistoleer_we.tga,0,0,77,60,0
icon_unit_grenadier_am,icon_unit_grenadier_am.tga,0,0,77,60,0
icon_unit_grenadier_fe,icon_unit_grenadier_fe.tga,0,0,77,60,0
icon_unit_grenadier_me,icon_unit_grenadier_me.tga,0,0,77,60,0
icon_unit_grenadier_we,icon_unit_grenadier_we.tga,0,0,77,60,0
icon_unit_incendiarymortar_am,icon_unit_incendiarymortar_am.tga,0,0,77,60,0
icon_unit_incendiarymortar_fe,icon_unit_incendiarymortar_fe.tga,0,0,77,60,0
icon_unit_incendiarymortar_me,icon_unit_incendiarymortar_me.tga,0,0,77,60,0
icon_unit_incendiarymortar_we,icon_unit_incendiarymortar_we.tga,0,0,77,60,0
icon_unit_mortarman_am,icon_unit_mortarman_am.tga,0,0,77,60,0
icon_unit_mortarman_fe,icon_unit_mortarman_fe.tga,0,0,77,60,0
icon_unit_mortarman_me,icon_unit_mortarman_me.tga,0,0,77,60,0
icon_unit_mortarman_we,icon_unit_mortarman_we.tga,0,0,77,60,0
icon_unit_bombletmortar,icon_unit_bombletmortar.tga,0,0,77,60,0

icon_unit_archer,icon_unit_archer.tga,0,0,60,60,0

// HEAVY INFANTRY 1,,,,,
icon_unit_assaulttriflemen_am,icon_unit_assaulttriflemen_am.tga,0,0,77,60,0
icon_unit_assaulttriflemen_fe,icon_unit_assaulttriflemen_fe.tga,0,0,77,60,0
icon_unit_assaulttriflemen_me,icon_unit_assaulttriflemen_me.tga,0,0,77,60,0
icon_unit_assaulttriflemen_we,icon_unit_assaulttriflemen_we.tga,0,0,77,60,0
icon_unit_maceman_am,icon_unit_maceman_am.tga,0,0,77,60,0
icon_unit_maceman_fe,icon_unit_maceman_fe.tga,0,0,77,60,0
icon_unit_maceman_me,icon_unit_maceman_me.tga,0,0,77,60,0
icon_unit_maceman_we,icon_unit_maceman_we.tga,0,0,77,60,0
icon_unit_manatarms_am,icon_unit_manatarms_am.tga,0,0,77,60,0
icon_unit_manatarms_fe,icon_unit_manatarms_fe.tga,0,0,77,60,0
icon_unit_manatarms_me,icon_unit_manatarms_me.tga,0,0,77,60,0
icon_unit_manatarms_we,icon_unit_manatarms_we.tga,0,0,77,60,0
icon_unit_musketeer_am,icon_unit_musketeer_am.tga,0,0,77,60,0
icon_unit_musketeer_fe,icon_unit_musketeer_fe.tga,0,0,77,60,0
icon_unit_musketeer_me,icon_unit_musketeer_me.tga,0,0,77,60,0
icon_unit_musketeer_we,icon_unit_musketeer_we.tga,0,0,77,60,0
icon_unit_objectiveforcewarrior,icon_unit_objectiveforcewarrior.tga,0,0,77,60,0
icon_unit_regular_am,icon_unit_regular_am.tga,0,0,77,60,0
```

Now you should enter the exact same line into the file `dbsprite_unitIcons.csv`  
Located in:  
`EE2X_db.zip\UI\ UnpackedSprites\dbsprite_unitIcons.csv`

There is one more thing left, you need to assign the unit a new icon. For example, let's change the Roosevelt icon to a new one. Open the file that has the Roosevelt defining text block - this is `american_campaign_units.ddf`.

Located in: `american_campaign_units`  
`EE2X_db.zip\EE2X_db\Units\american_campaign_units.ddf`

The block itself looks like this:

```

//-----
// AMERICAN 1 - TEDDY ROOSEVELT
//-----
unitType LeaderRoosevelt
{
    parent = LeaderMilitary
    properties {
        displayName = tx_utm_LeadTedRoos_name
        displayNameScenEdit = tx_utm_LeadTedRoos_sname
        hitpoints = 3000
        verboseTooltip = vtt_unit_LeadTedRoos
    }
    sounds {
        sndCmdMove = ["v_leader_mil_02_com01" "v_leader_mil_02_com02"]
        sndCmdAttack = ["v_leader_mil_02_com01" "v_leader_mil_02_com02" "v_leader_mil_02_spe01"]
        sndSelection = ["v_leader_mil_02_sel01"]
    }
    abilities = [
        Attack { range = 5.0; reloadTime = 1; applyDamageTime = 0.5; damage = 40; weaponType = LightProjectile }
        LOS { range = 5 }
        AreaEffect {
            effects = [
                HighFervor
                HighMorale
                Discipline
            ]
        }
        SpecialPower {
            effects = [
                FireAndManeuver
                Rally
            ]
        }
    ]
    icon = icon_unit_roosevelt
}

unitModel LeaderRoosevelt
{
    scaleType = kScaleUnit
    Parent = BaseHuman
    defaultModel = camp_am11_roosevelt.nif
    ForceScale = 0.260
    childNames = [ UnitShadow Beam_Tracer01 Muzzle_Rifleman ]
    states = [
        {
            StateName = Idle AnimName = camp_am11_idle.kf
            AnimVariants = [ camp_am11_idle2.kf ]
            AnimVariantsWeights = [ 0.05 ]
        }
        {
            StateName = walk AnimName = camp_am11_walk.kf AnimSpeedScale = 2.25 }
        {
            StateName = Run AnimName = camp_am11_jog.kf
        }
        {
            StateName = Attack1 AnimName = camp_am11_attack.kf
            TextKeys = [
                { Name = PLAY_SOUND Data = fx_rifle_fire KeyTime = 0.35 }
            ]
        }
        {
            StateName = Death AnimName = camp_am11_death.kf
        }
    ]
    UIRenderingParams {
        xoffset = 0.01
        yoffset = 0.32
        zoffset = -6.48
        scale = 2.43
        angle = 23.68
    }
}

```

A line in the block

icon = icon\_unit\_roosevelt

change to

icon = icon\_unit\_yourunitname

Save all changes.

All is ready. Now we're starting the game and looking at the new Roosevelt icon.

## 6 How to add a new 3D vehicle or building model to the game.

Basics of work in the program NifSkope.

(To work you must have the model Volkshalle.nif and Volkshalle.tga These files must be in the archive Volkshalle11.rar If you do not have this archive you can download it here:

<http://ee.heavengames.com/downloads/showfile.php?fileid=2835>

You must also have NifSkope installed. You can download it here:

<http://niftools.sourceforge.net/wiki/NifSkope> )

Let's get acquainted with the amazing program NifScope.

*author of the article IgorLutiy (Igor\_Ra)*

//

### Subtlety of working with NifSkope

#### Introduction

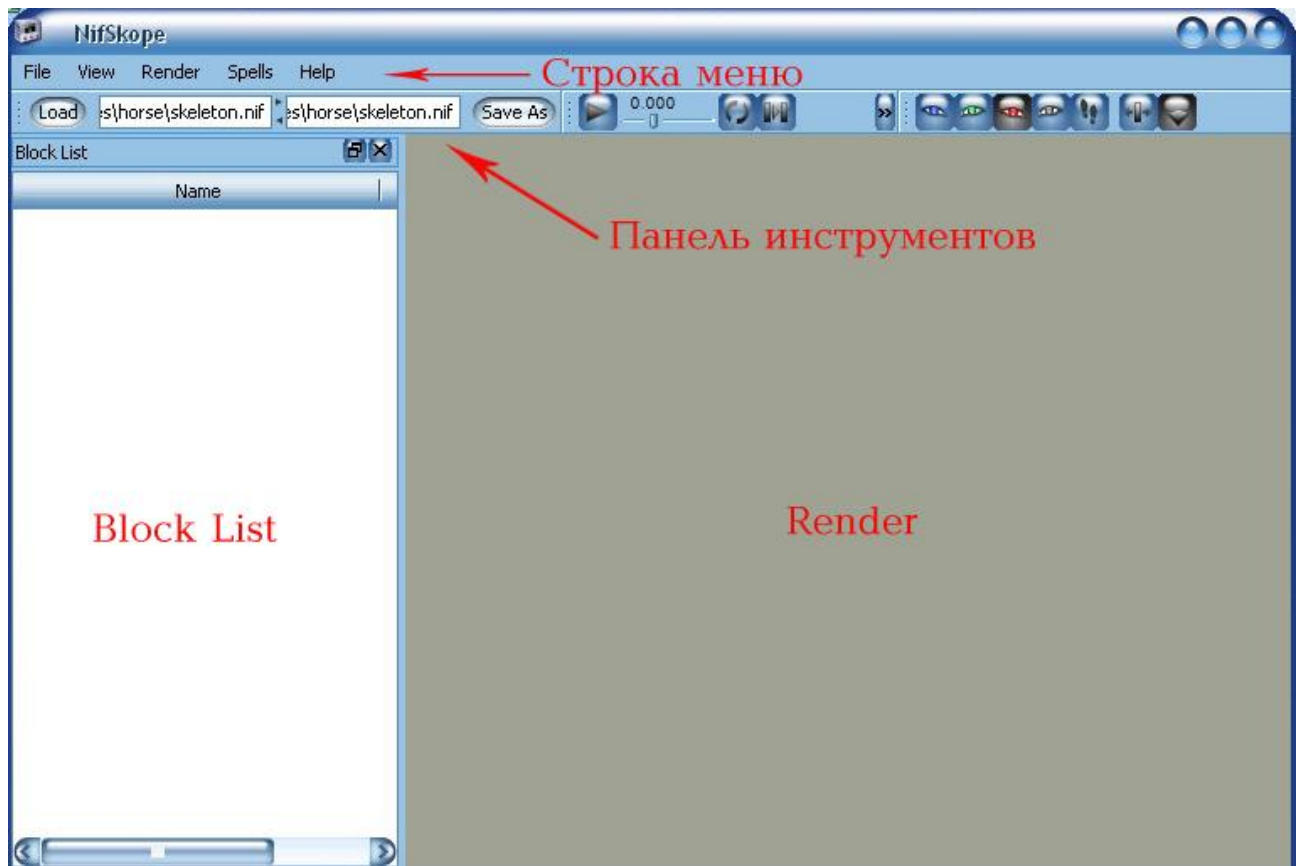
In this article, we'll look at the basics of working with the program **NifSkope**, which allows you to open and edit files saved in **NetImmerse File Format (Nif)**. This format is used to store 3D models in games based on the popular game engine **NetImmerse/Gamebryo**. These are games like **TES3: Morrowind**, **TES4: Oblivion**, **Civilization IV**, **Empire Earth II**, **Empire Earth III**, **The Guild 2**, **Axis and Allies** (and many others).

I'm going to rely on using the program for models from game **TES4: Oblivion**.

I have version 1.0 installed so if you are using a different version of the program, maybe something is not will fit. But in general I think everything will be clear anyway. What can we do with this program? Change the texture of the model, work with collisions, change skeleton, see animation and much more ... Let's start by getting to know the interface.

#### 1. Interface

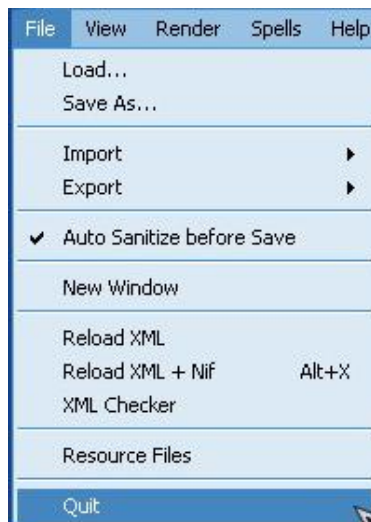
We open the program **NifSkope**. We see this window:



What do we see? Menu bar, toolbar and two windows – **Block List** (block list) and **Render** (Window).

### Menu bar

Let's start by looking at the menu bar. It consists of five parts: **File**, **View**, **Render**, **Spells**, **Help**.



### File

**Load...** – Uploading the file. Extensions that are supported: .Nif, .kf, .kfa, .kfm.

**Save As...** – save as ... Save the file by naming it.

**Import** – import from formats: .3ds, .obj

**Export** – export to .obj

**Auto Sanitize before Save** – automatic processing before saving

**New Window** – opens a new program window

**Reload XML** – this and the next two menu items are related **XML** (it means eXtensible Markup Language – Extensible Markup Language). We won't touch it.

**Reload XML + Nif**

**XML Checker**

**Resource Files** – something related to archives **BSA**. What is it for - I do not understand ☺

**Quit** - Quit from program



## View

**Interactive Help** – online help opens from the bottom in the inline window

**Block List** – shows a list of blocks

**Block Details** – opens a built-in window below with a detailed description of blocks

**KFM**

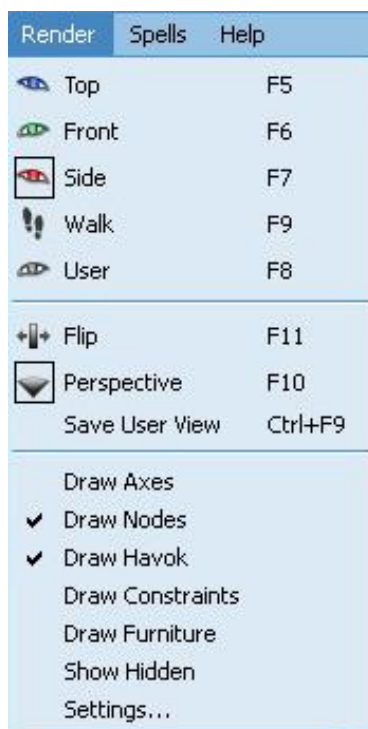
**Toolbars** – enables the display of buttons on the toolbar. Group of buttons **Load**

**Save, View, Animation**

**Show Blocks in Tree** – displays the block list as a tree

**Show Blocks in List** – displays the block list as a list

**Select Font...** - font setting



## Render

**Top** – view from above

**Front** – front view

**Side** – side view

**Walk** – view «from within»

**User** – custom view (contains a view saved with the command **Save User View**, which is below)

**Flip** – mirror reflection

**Perspective** – perspective

**Save User View** – preserves the user's view

**Draw Axes** – enables display of the coordinate system

**Draw Nodes** – includes showing nodes

**Draw Havok** – enables display of collisions

**Draw Constraints** – I did not notice the effect

**Draw Furniture** – I did not notice the effect

**Show Hidden** – I did not notice the effect

**Settings...** - program settings (an important point that we will come back to later)



## Spells

**Animation** – allows you to load an animation for viewing

**Block** – includes two teams: **Insert** (insert block) and **Remove By id** (remove block by **id**)

**Optimize** – optimization

**Sanitize** – treatment

**Batch** – package



## Help

**Interactive Help** – online help opens from the bottom in the inline window of **NifSkope**

**Documentation & Tutorials** – opens a documentation page and tutorials for **NifSkope**

**NifSkope Help & Bug Report Forum** – opens the forum page **NifTools**

**NifTools Wiki** – opens page **Wiki** by **NifTools**



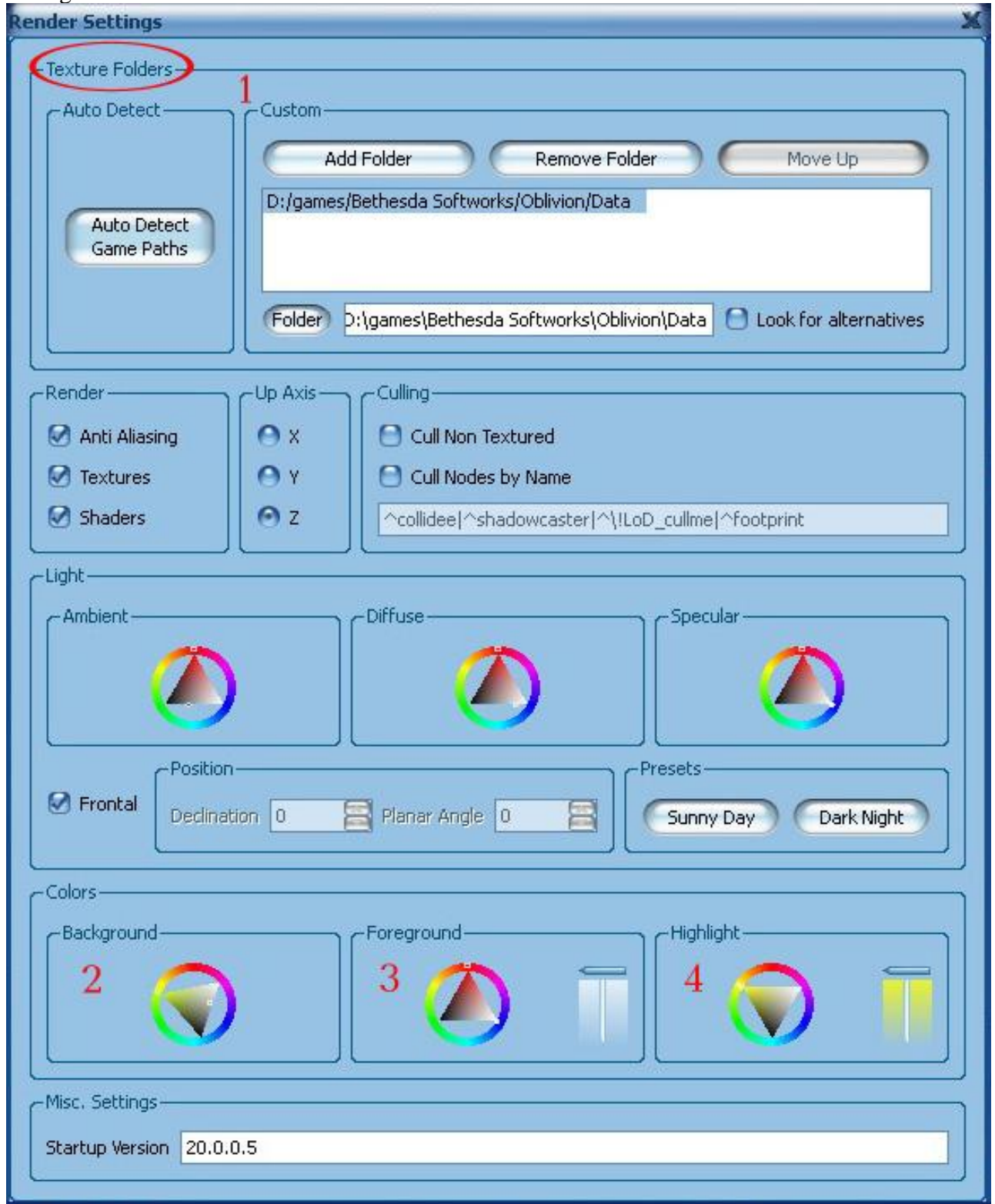
**NifTools Downloads** – opens a page for downloading tools for working with **NifSkope**

**About Qt** – about the program with which this tool was created

**About NifSkope** – about the program

### Settings

Separately, mention should be made of **Settings (Render>Settings)**, it is about program settings:



1. The top field **Texture Folders**. This is a very important matter. Here we define the path for textures. This is a specific location on your computer. **NifSko**pe looking up textures referenced in the NIF file.

To specify the path, press the button **Add Folder**. In the track list texture search, you will see a new line that looks like this **Choose A Folder**. Make sure the line is highlighted when you take the next step. There are two ways to edit a track in the list - manually, by double clicking or clicking the button **Folder** and you can enter the browser to determine the location on the computer.

You can reveal as many search paths as you want, but preferably one of them. In Oblivion, typical **NIF** the file will contain a path to textures such as "**textures\weapons\IronDagger.dds**" This means that the folder in this path is one above the "textures" folder. In the case of a normal Oblivion installation this would be the folder **Data**, which should be set for the search path.

**Field Colors:**

2. **Background** – background color in the preview window

3. **Foreground** – color of nodes in the display window (default is white). The slider on the right side of the palette determines the degree of transparency.

4. **Highlight** – the color of the selected node (yellow by default). The slider on the right side of the palette determines the degree of transparency.

### Toolbar

And now the toolbar. It basically repeats the commands on the menu bar, but there are those that none (related to animation).



1. **Load...** – Upload a file

2. Shows the path to the loaded file

3. Path to the file you save via **Save As...**

4. **Save As...** – Save the file as...

5. Button **Play** – is used to play the animation

6. Animation scroll bar

7. **Loop** – loops showing animation. It only reflects on the scroll bar

8. **Switch** – animation switch. If you have loaded some animations, then when the pressed button will play all animations in a row. When not pressed, only the current one.

9. This group of buttons repeats the first seven commands on the menu **Render**

## 2. Textures and NifSko

Working with textures in NifSko takes an important and significant place. Therefore, we will first consider it carefully.

Let's open the horse model. In the block list (**Block List**) and find the node **NiSourceTexture**

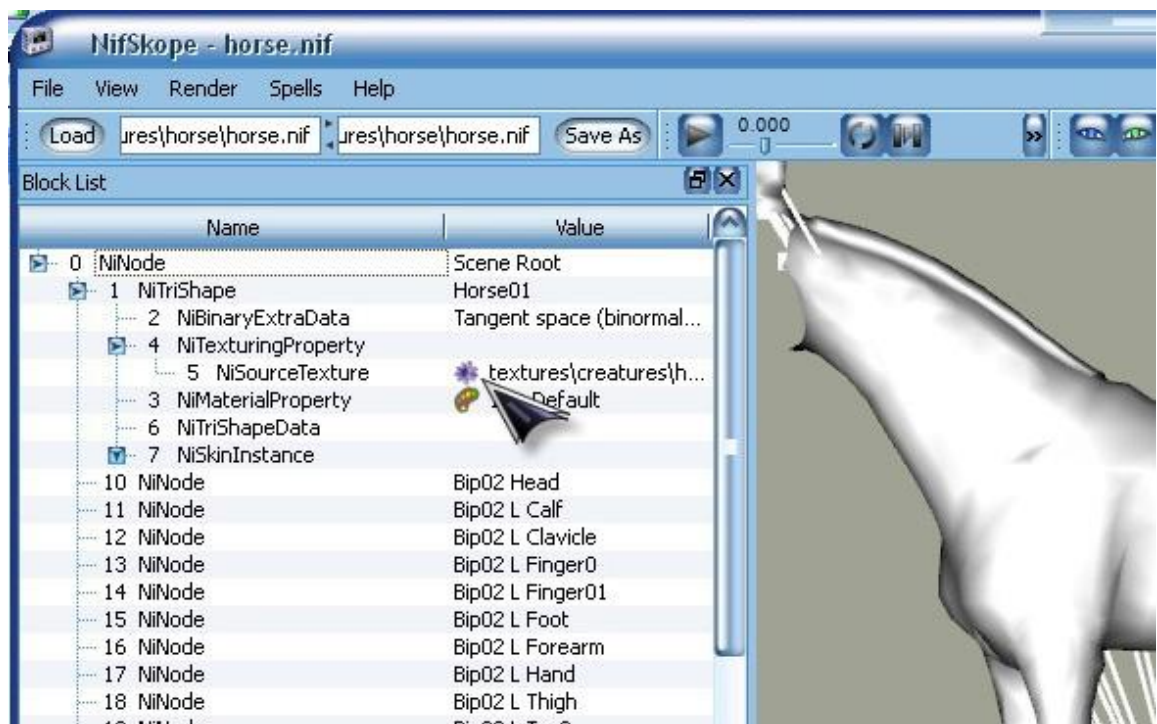
**NiNode>**

**NiTriShape (or NiTriStrips)>**

**NiTexturingProperty>**

**NiSourceTexture**

and on the right side we see a purple flower. Next to it, for example, is the Texture Path (for a horse) **textures\creatures\horse\Horse.dds** To view / edit a texture path, double-click on it with the left mouse button.



Don't forget that the path can be relative! It is not necessary to write:

D:\Game\Oblivion\Data\textures\creatures\horse\Horse.dds

you can write:

textures\creatures\horse\Horse.dds

D:\Game\Oblivion\Data – we skip this part. Otherwise, if you save the full path to the texture and another person installs the game to a different drive / folder - he will have problems displaying the texture! More specifically, it will not be displayed.

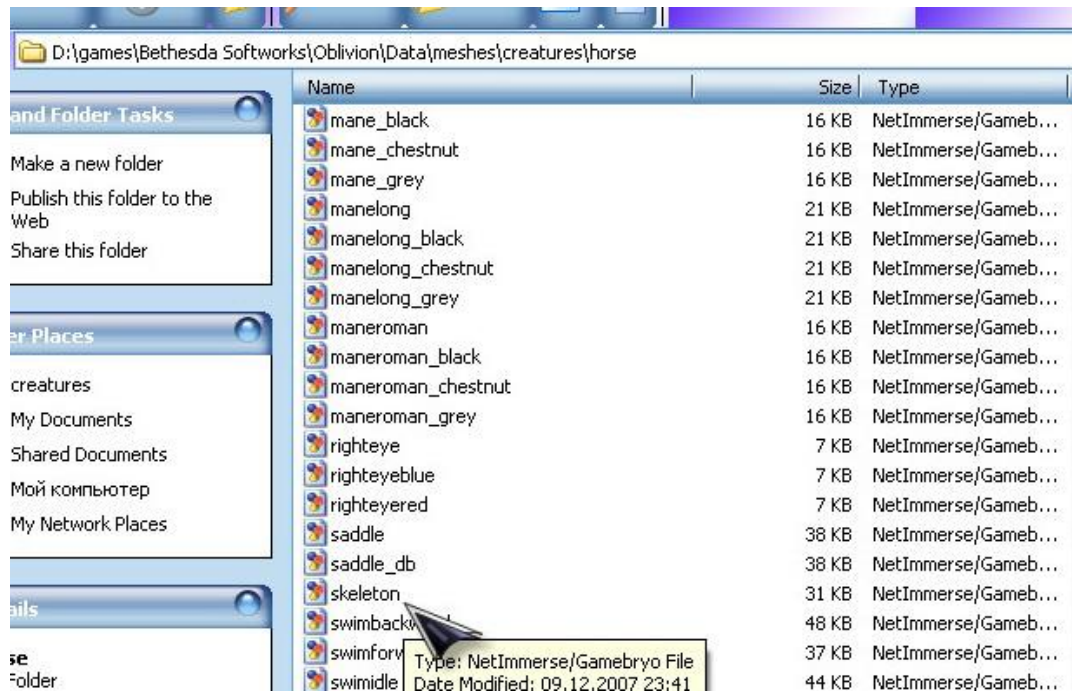
**PS:** 1. Remember to save the model with the new texture path.

2. Don't forget that some models may have more than one texture.

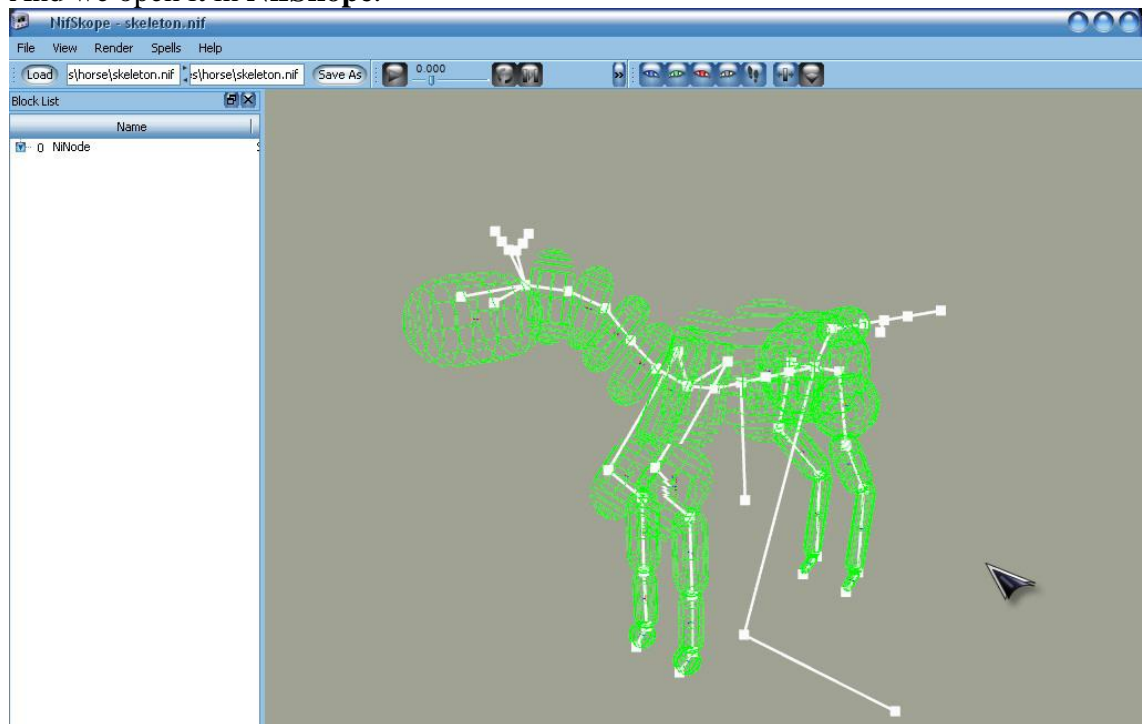
#### 4. Animation and NifSkope

There was once when I was interested in animation. The problem was, I didn't know how or where to edit. I asked on the forums, I thought - but how? Nothing and from anyone having achieved this, I concluded that you can watch them in **3Ds MAX**. But it was embarrassing. And 3D Max cannot stand in person. It turns out that the animation can be watched very easily in **NifSkope**. «Like me - you probably thought more than once - I tried to open animation files with this program many times and still nothing! It is impossible!" I, too, could not figure out how to do this for a long time. But everything turned out to be very simple, absurdly simple. And at the same time, it's very similar to how I opened the animations in 3D MAX.

So first we find the skeleton file we need. For example, For example I took a horse :



And we open it in NifSkope.

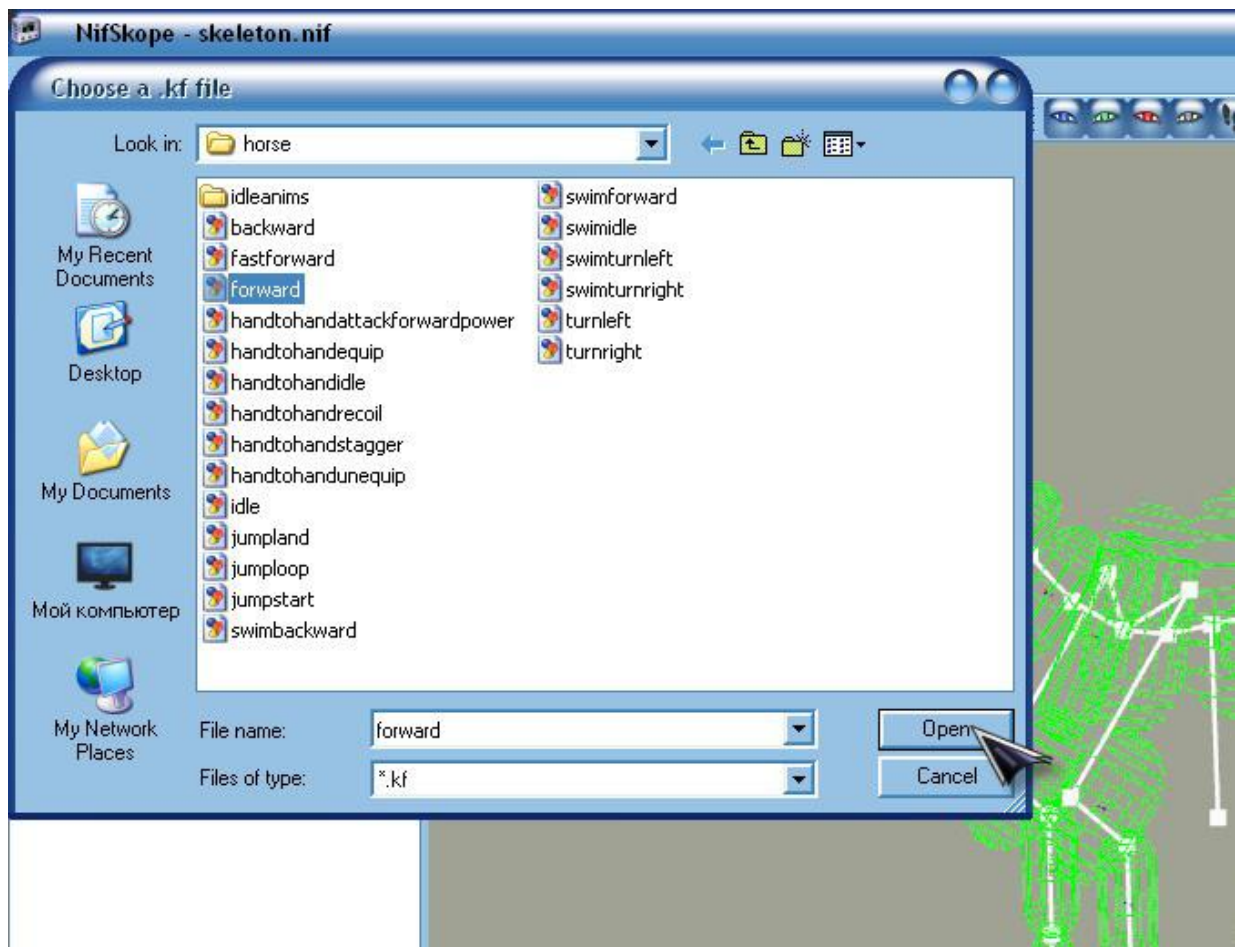


Now select on the menu bar **Spells>Animation>Attach.KF**

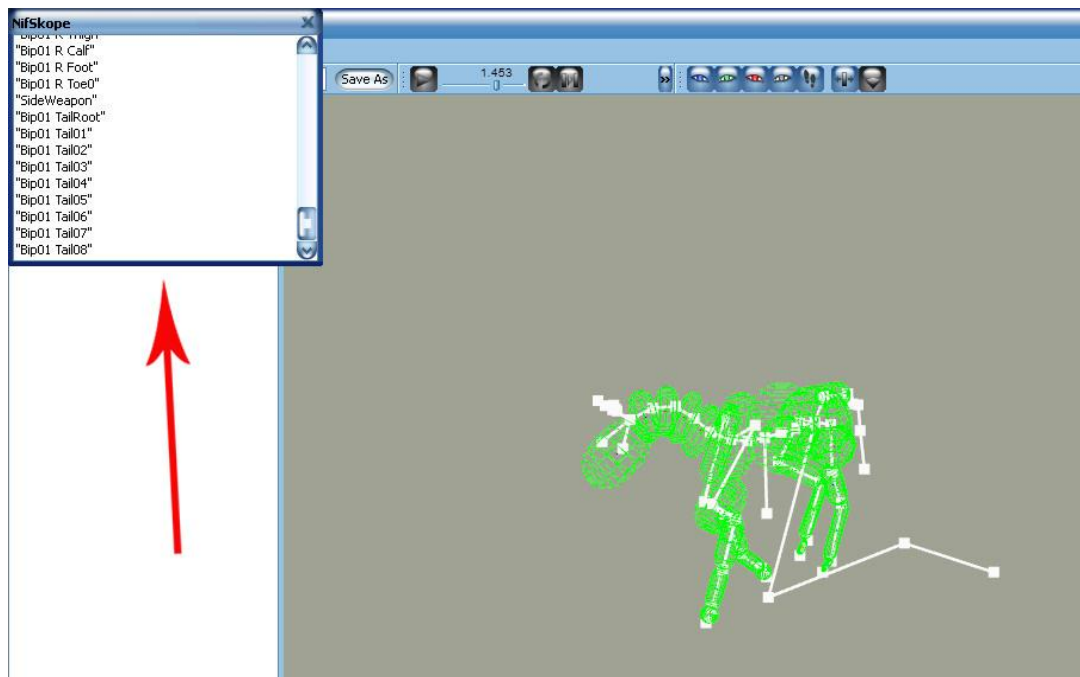




A window will open **Choose a .kf file** choose file (.kf), where you can select an animation file to attach to the armature. Let's choose and click the button **Open**.



Voila! The animation started to play. Perhaps a window like the one in the picture below will appear on the left side (it is indicated by a red arrow). Close them and ignore them. This does not affect the playback of the animation in any way. If the animation doesn't start to play, click the button **Play** on the toolbar.



////////////////////////////////////  
 More information about this program and the various tutorials (except for other games)  
 can be found here:

<http://rumor.ru/wiki/Категория:NifSkope>

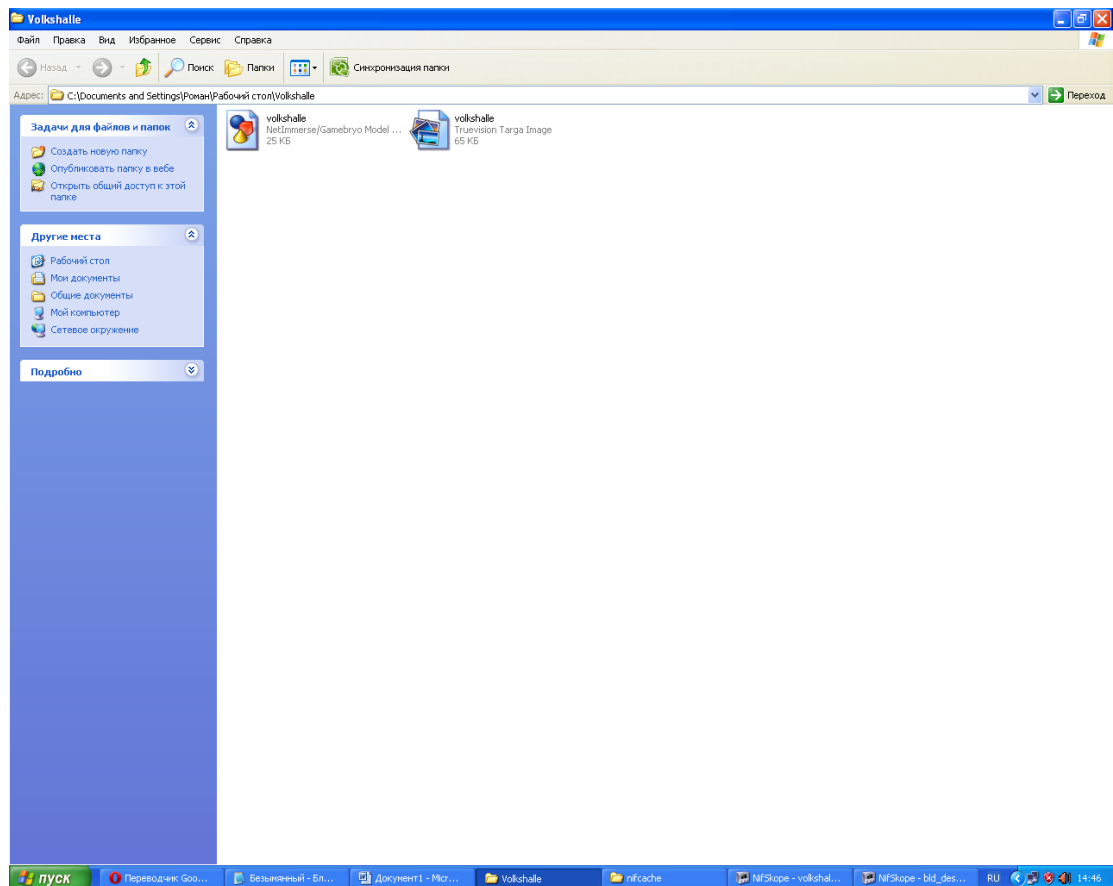
[http://niftools.sourceforge.net/wiki/NifSkope/Working\\_With\\_Nifs\\_101:\\_An\\_Introduction](http://niftools.sourceforge.net/wiki/NifSkope/Working_With_Nifs_101:_An_Introduction)

[http://niftools.sourceforge.net/wiki/NifSkope/Working\\_With\\_Nifs\\_101:\\_Basic\\_Use](http://niftools.sourceforge.net/wiki/NifSkope/Working_With_Nifs_101:_Basic_Use)

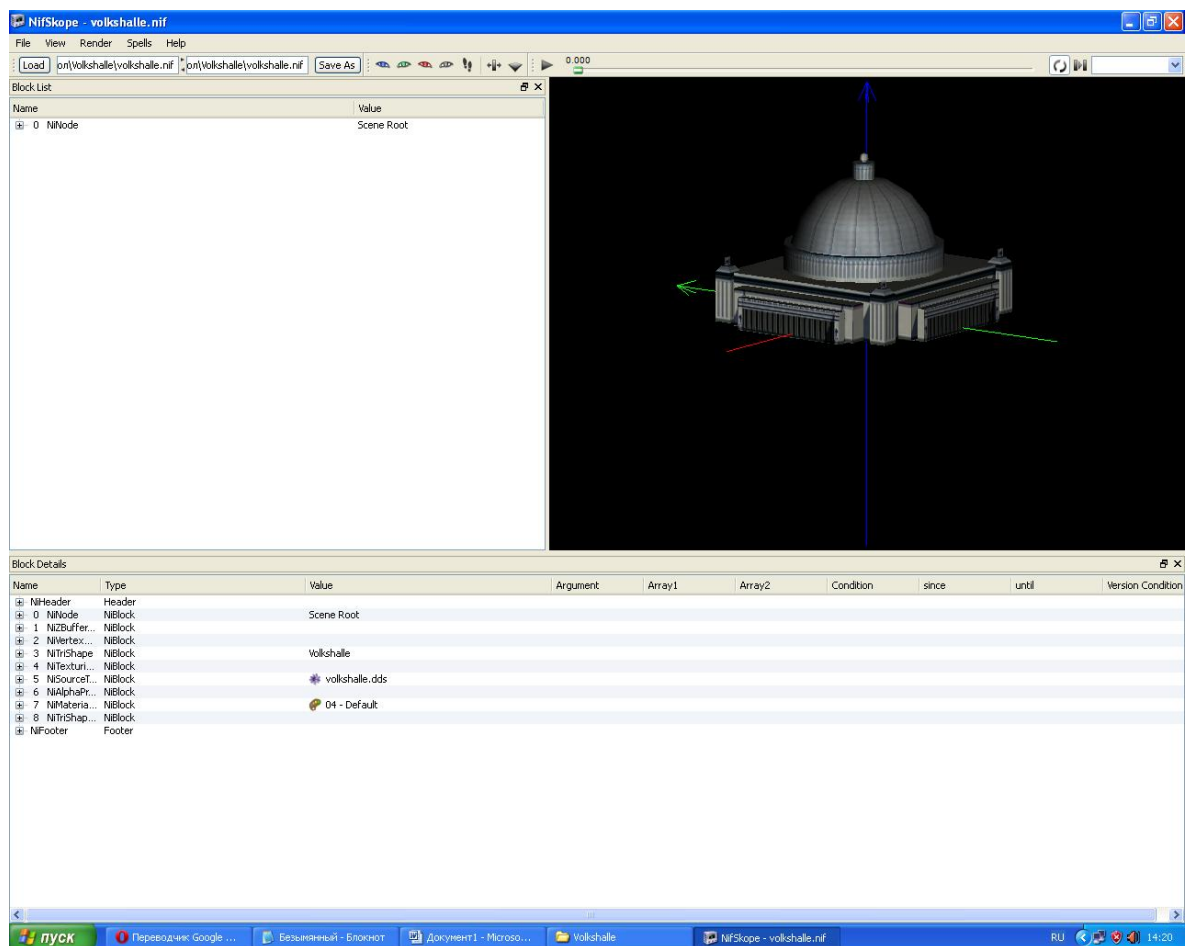
Now let's go directly to the ee2 building model customization process.

Open folder Volkshalle11

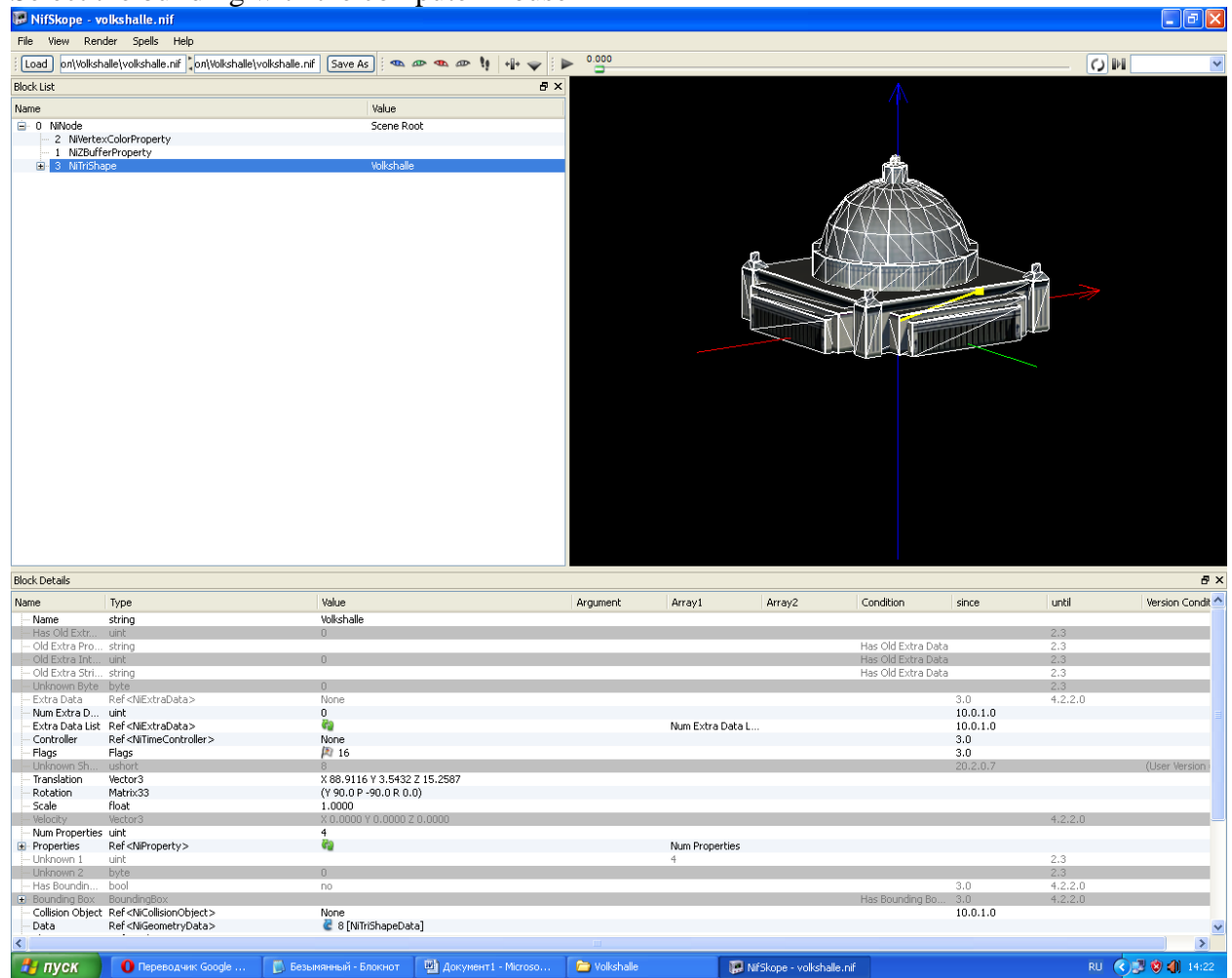




Open file Volkshalle.nif



Select the building with the computer mouse



Perform sequentially:

File

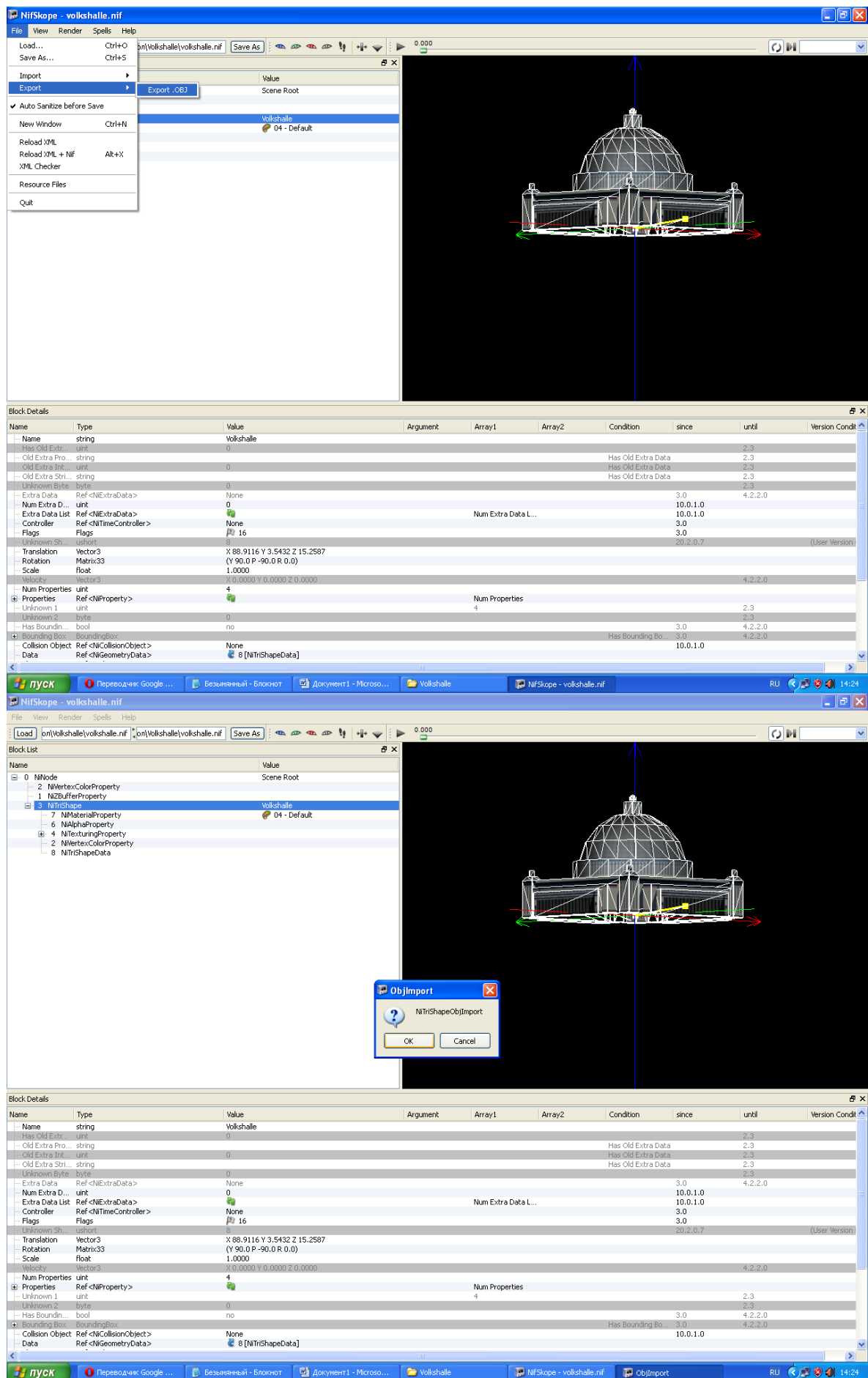
Export

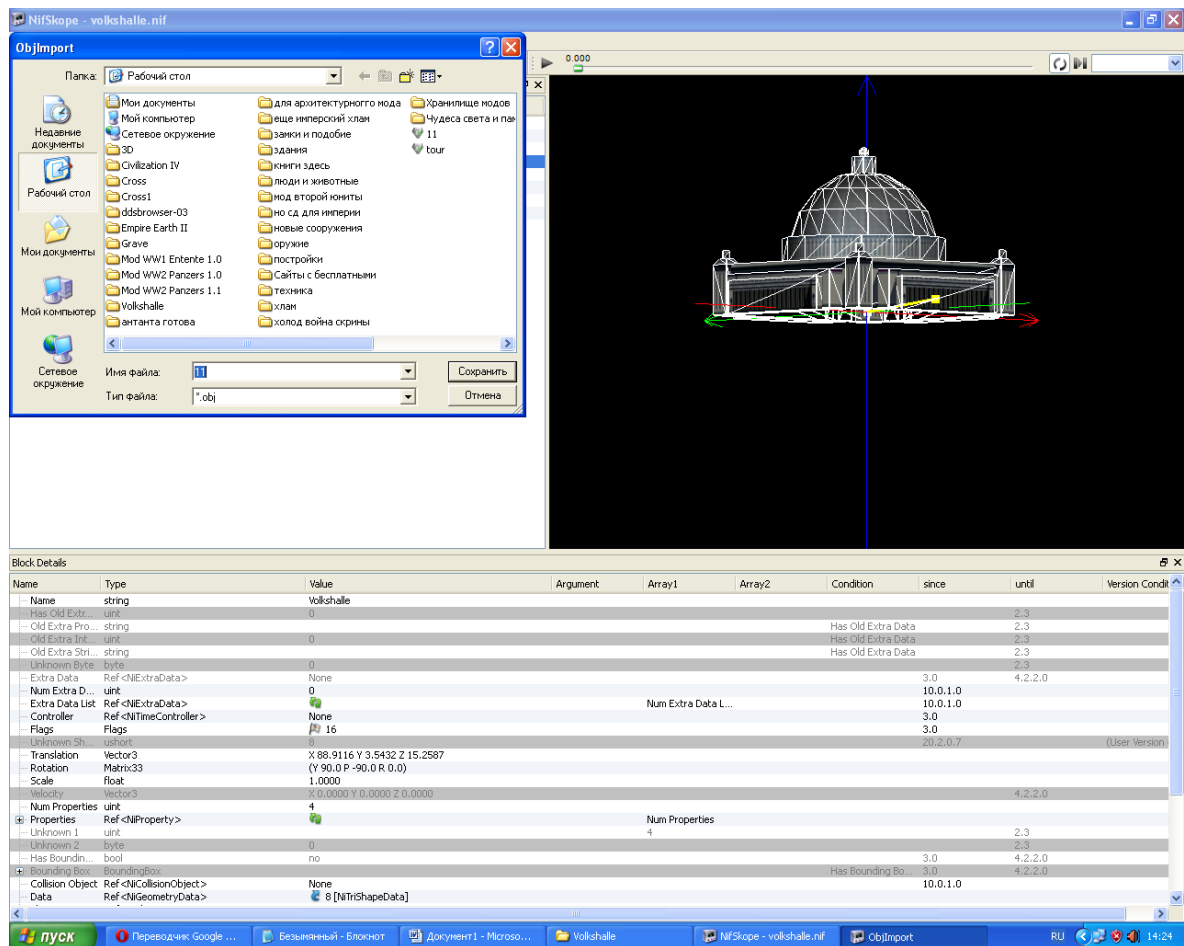
OBJ export

Save to your desktop.

File Name 11.

Let's name the file 11





Open the game folder, for example I have my Empire Earth II located in:

C: \ Program Files \ Sierra \ Empire Earth II

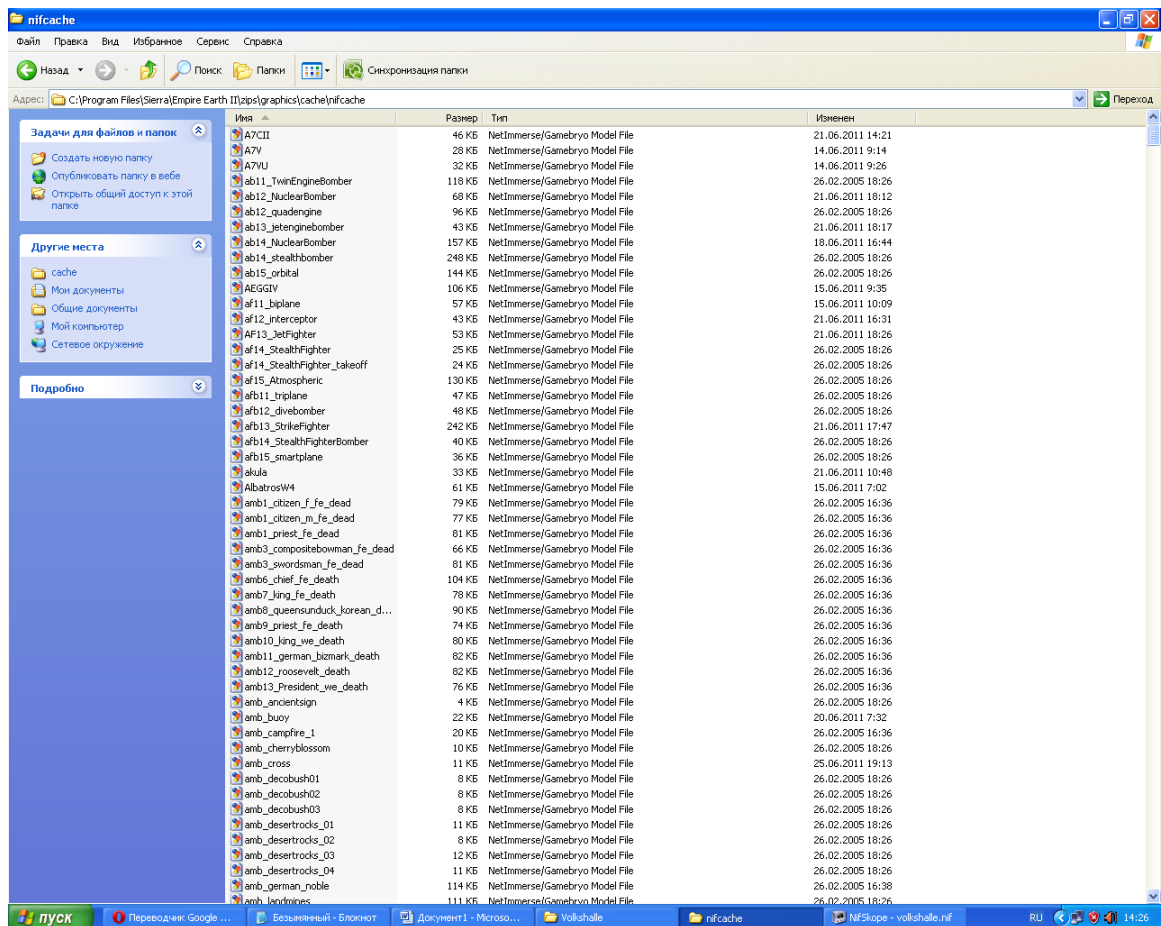
Find the archive named graphics.zip

C: \ Program Files \ Sierra \ Empire Earth II \ zips \ graphics.zip

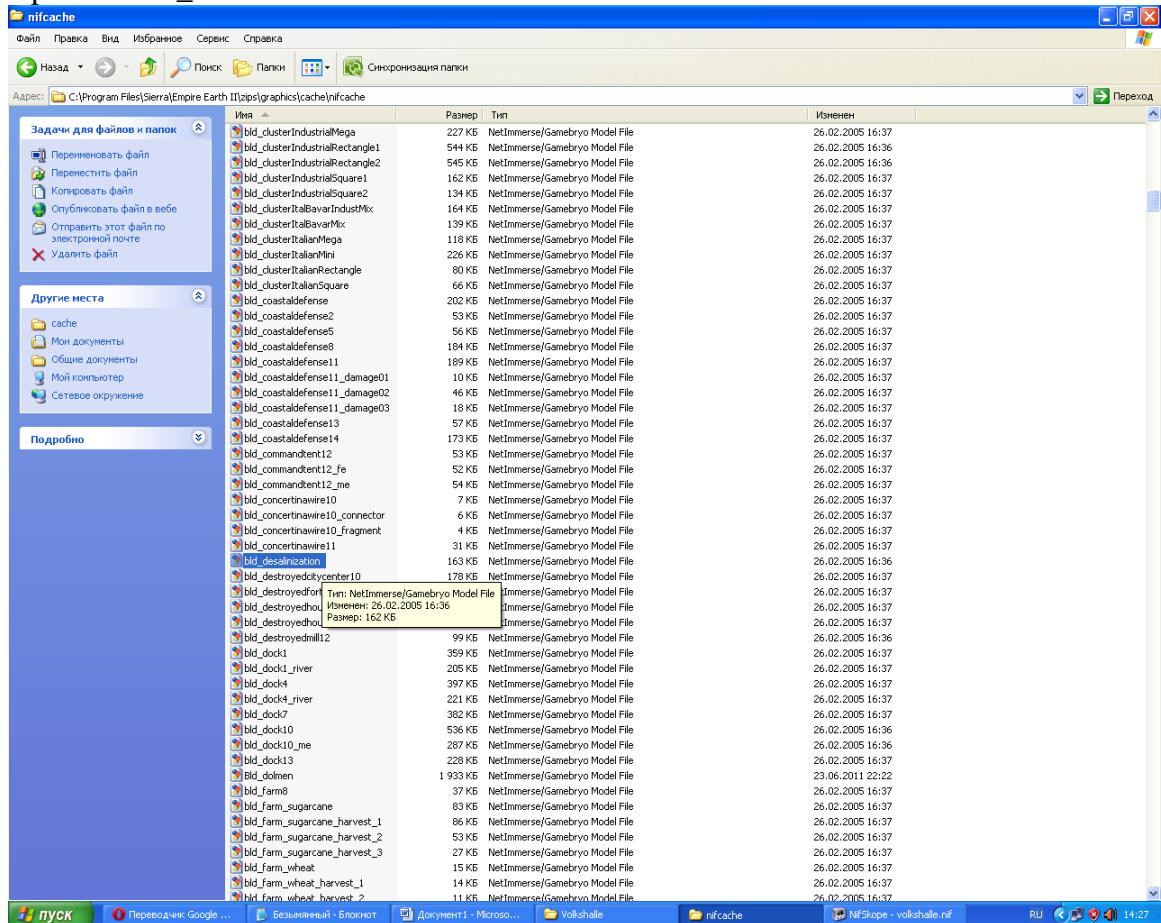
Unpack this archive, you should have a graphics folder.

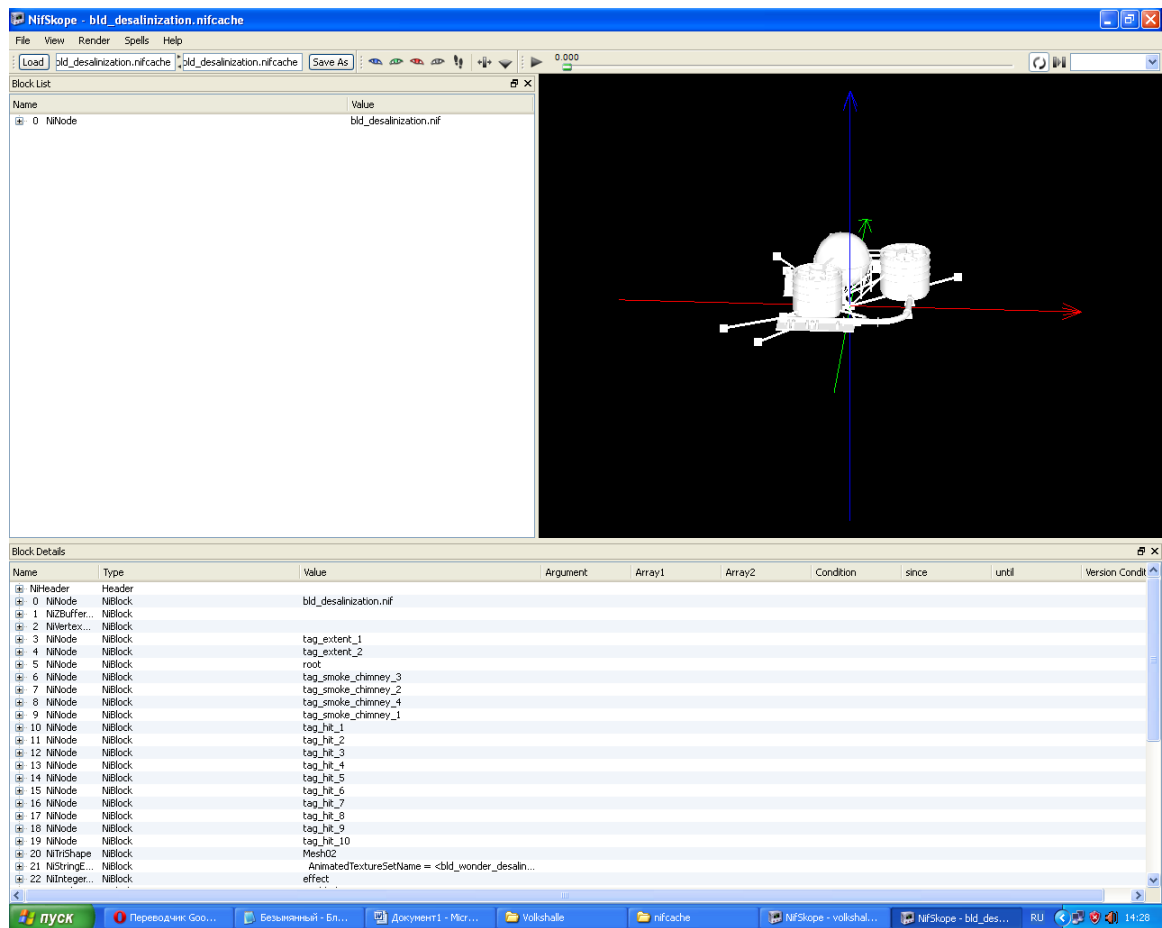
Now the archive can be deleted or moved to another place (for example to the desktop) Then open the nifcache folder, which is in:

(C: \ Program Files \ Sierra \ Empire Earth II \ zips \ graphics \ cache \ nifcache) This folder contains all the 3D models in the game.



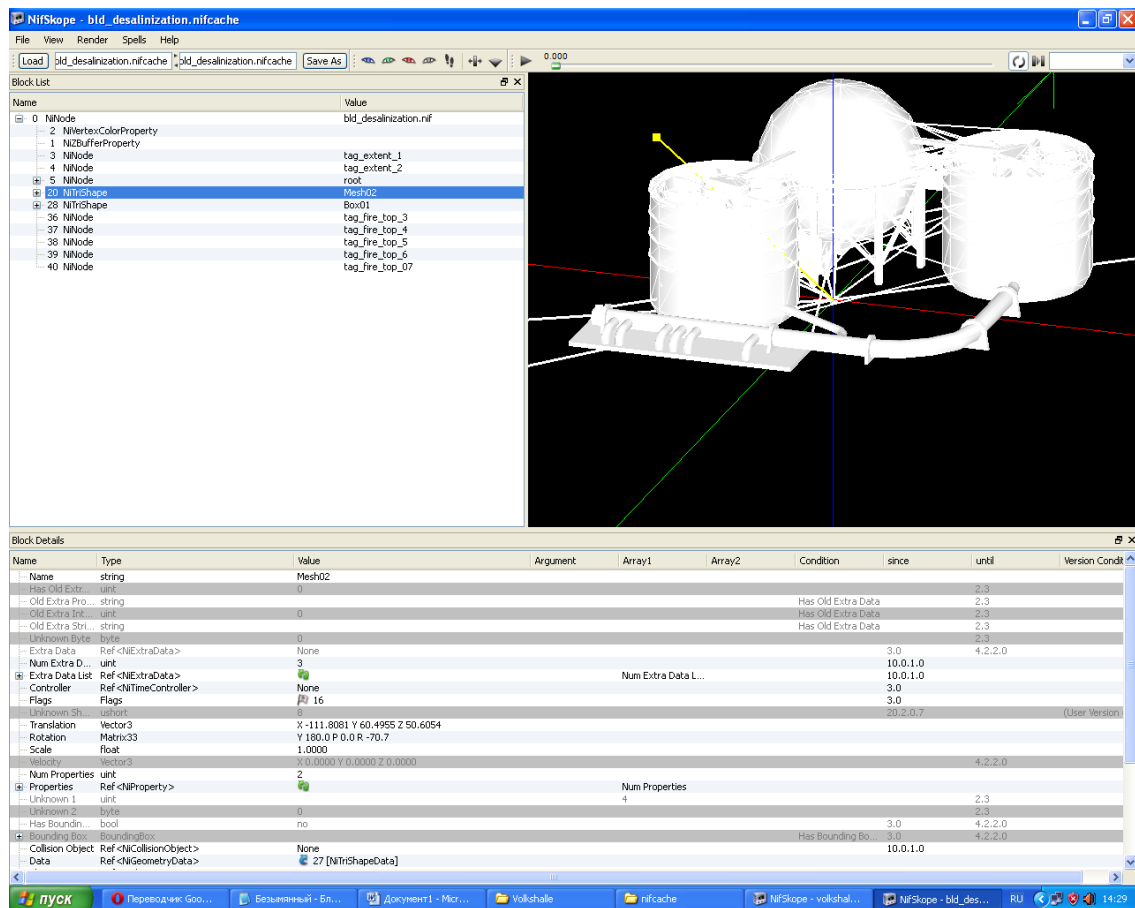
## Open file bld\_desalinization.nif



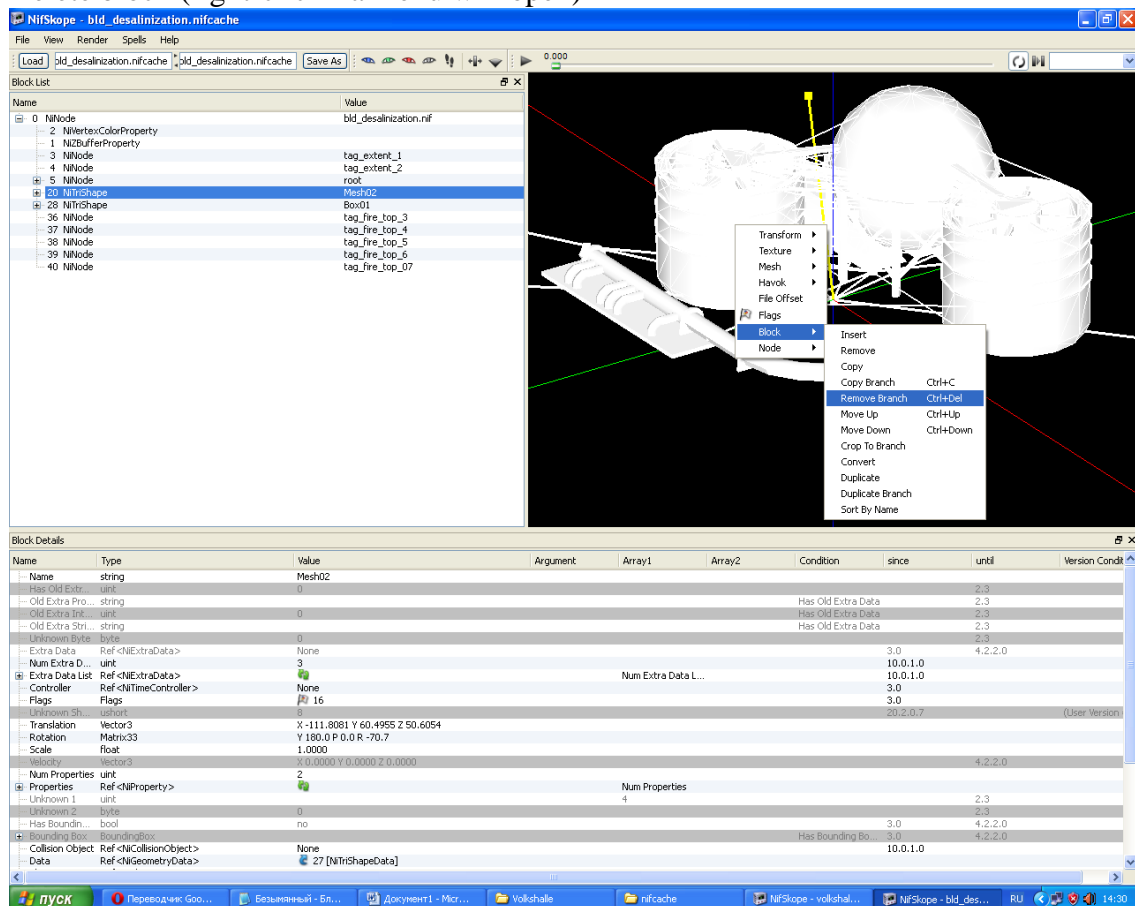


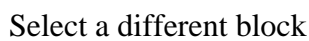
Select the block by selecting the building with the computer mouse



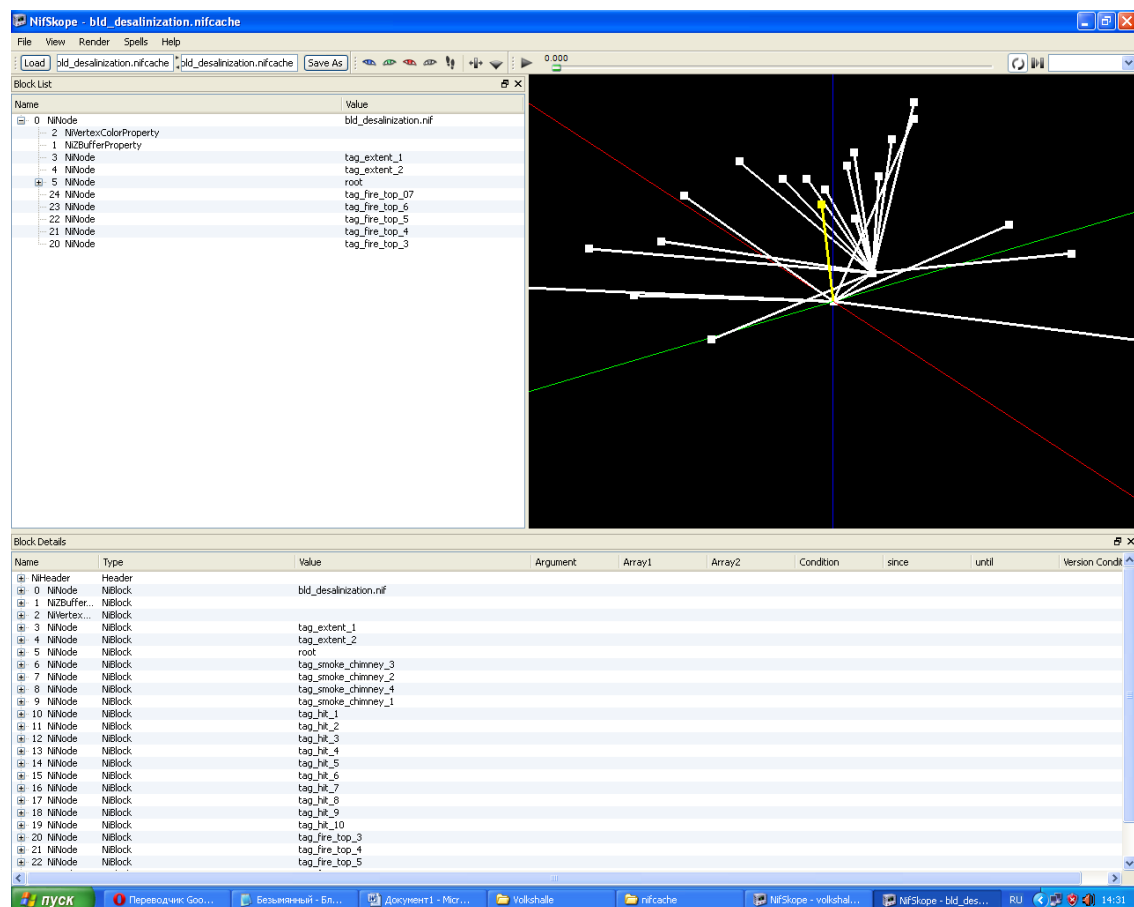
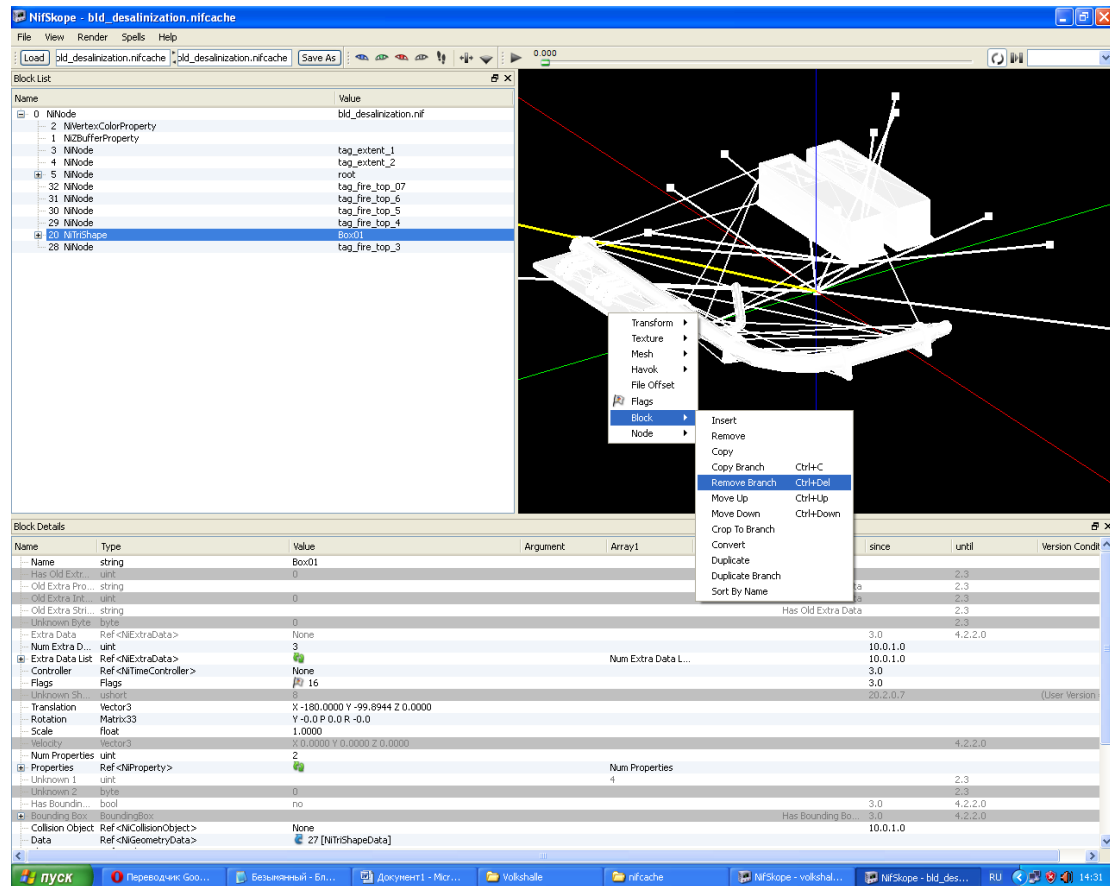


Delete block (right-click - a menu will open)

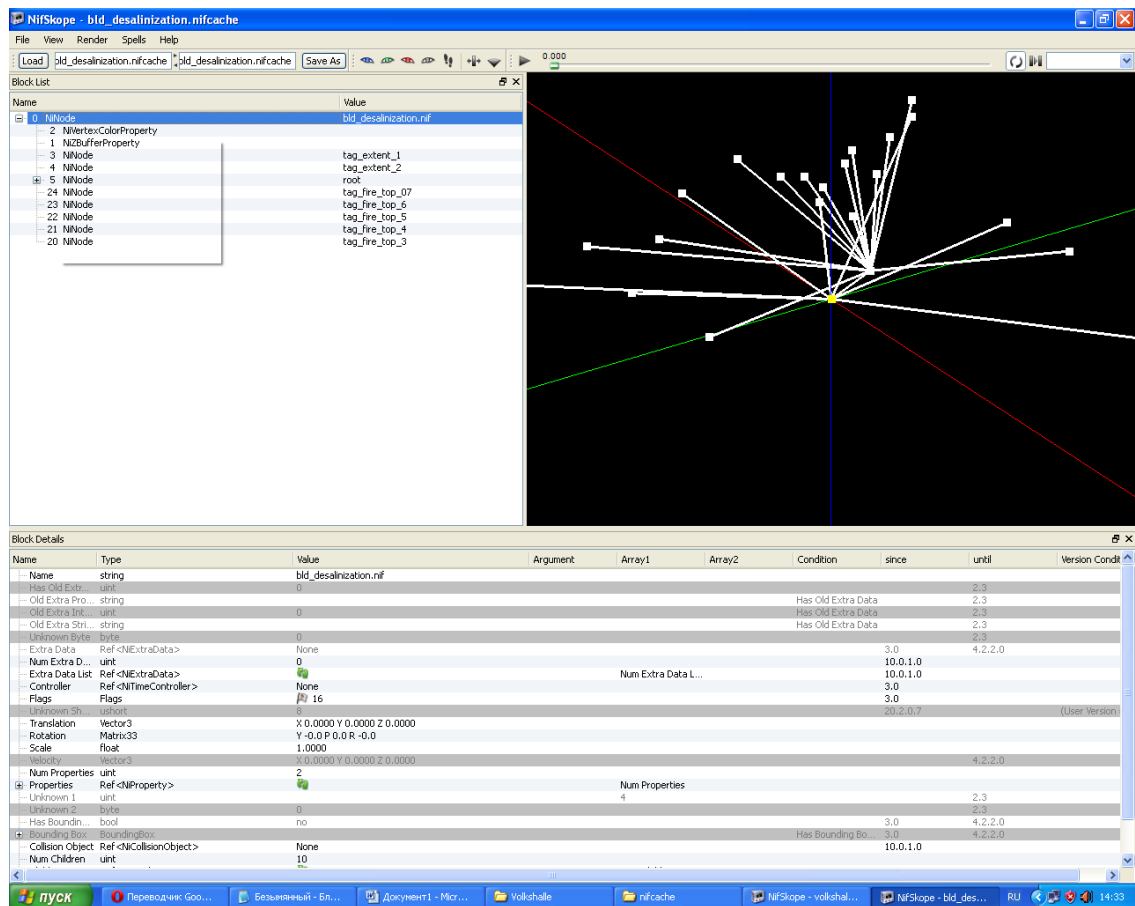




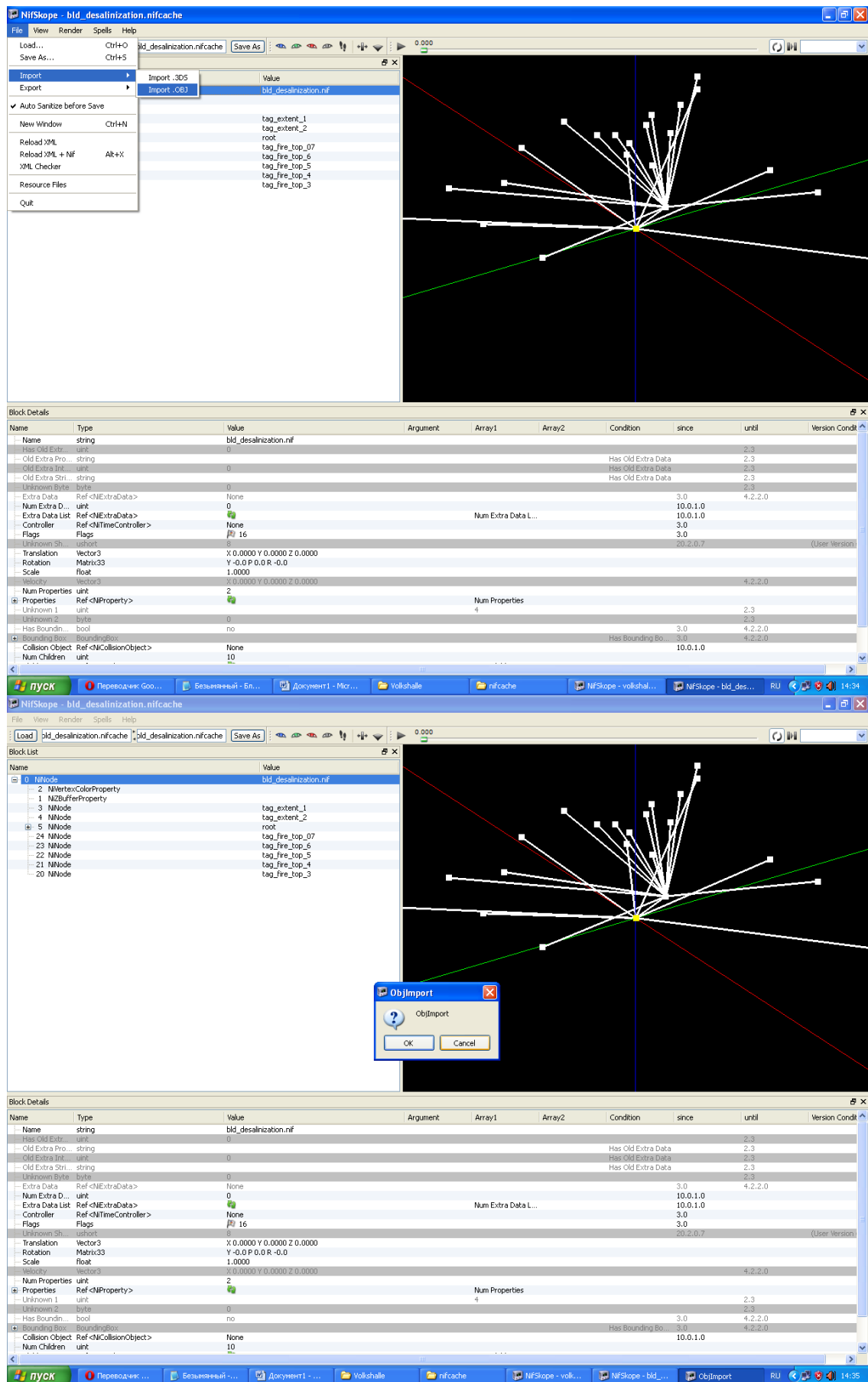
Let's remove the block

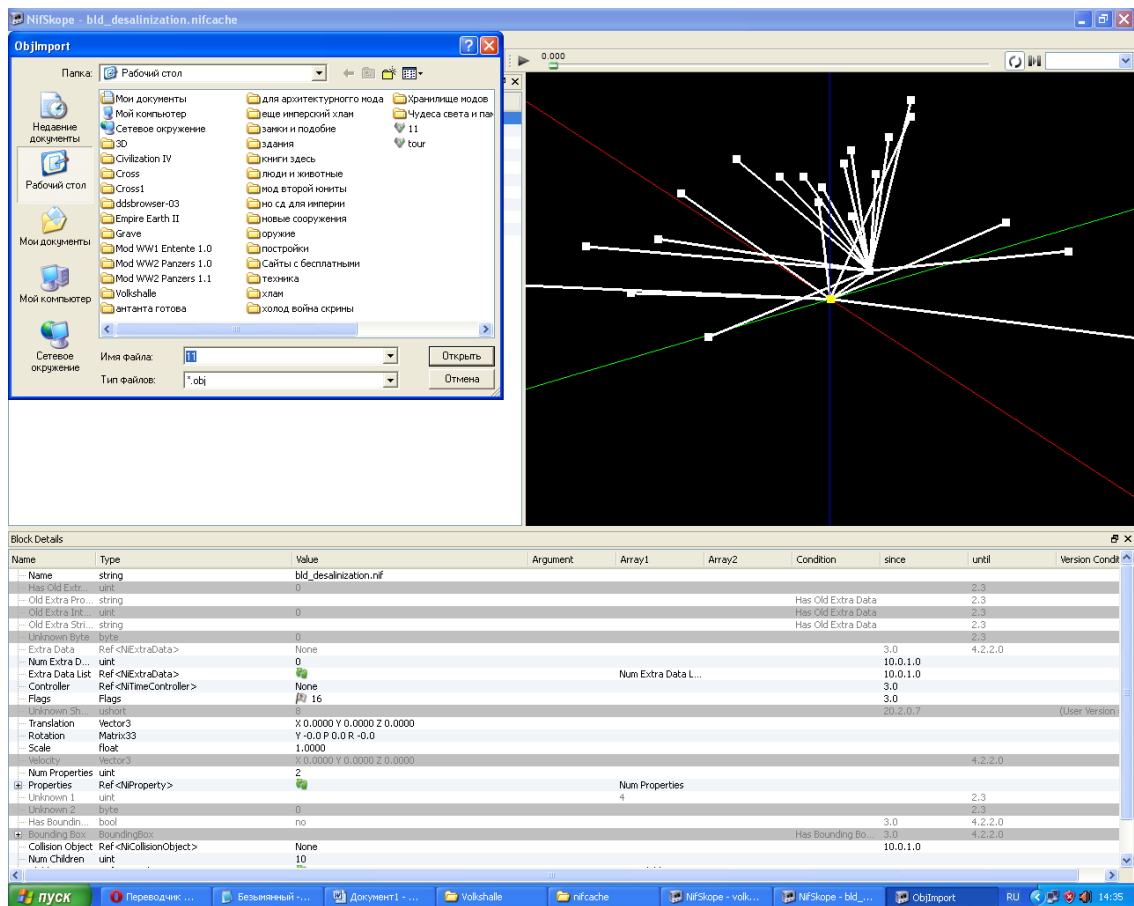


This will remove all blocks from the original model. Then select the topmost block NiNode



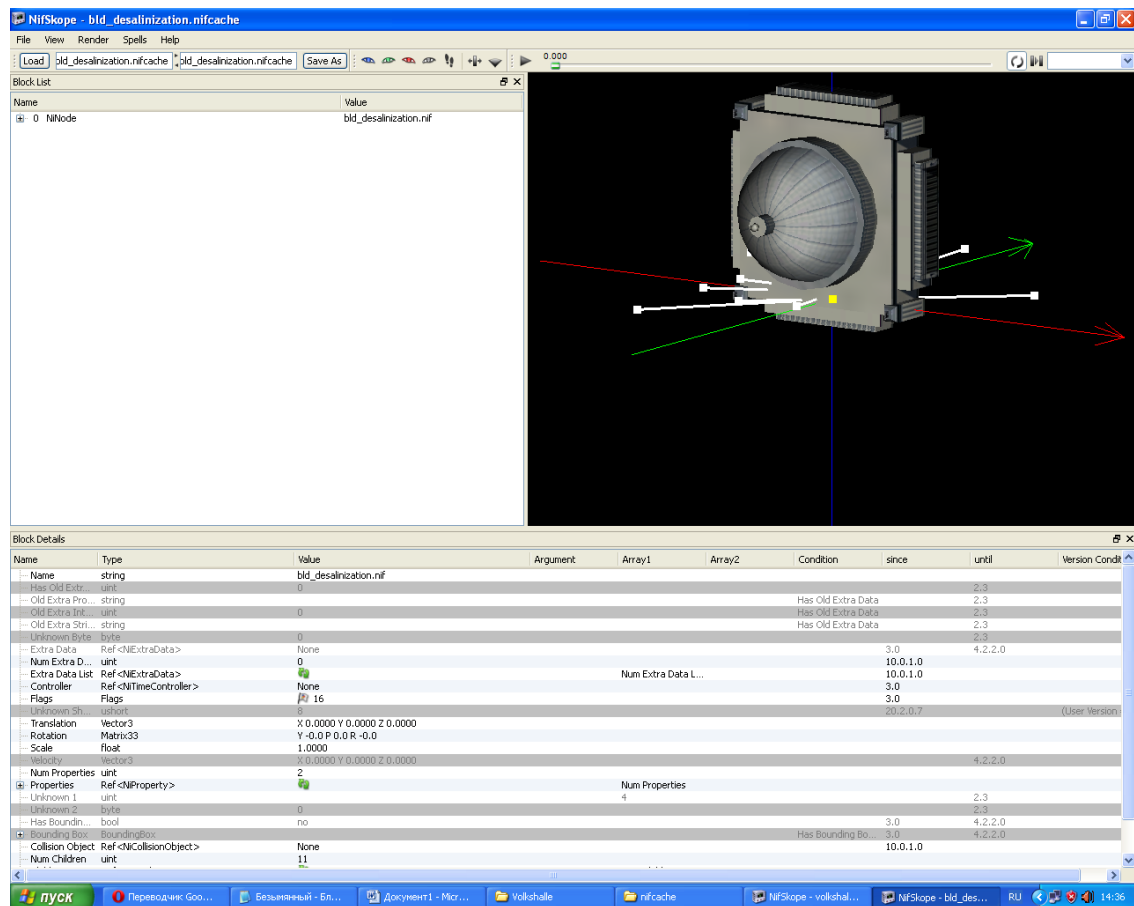
And then we import our new model  
select  
file  
OBJ import





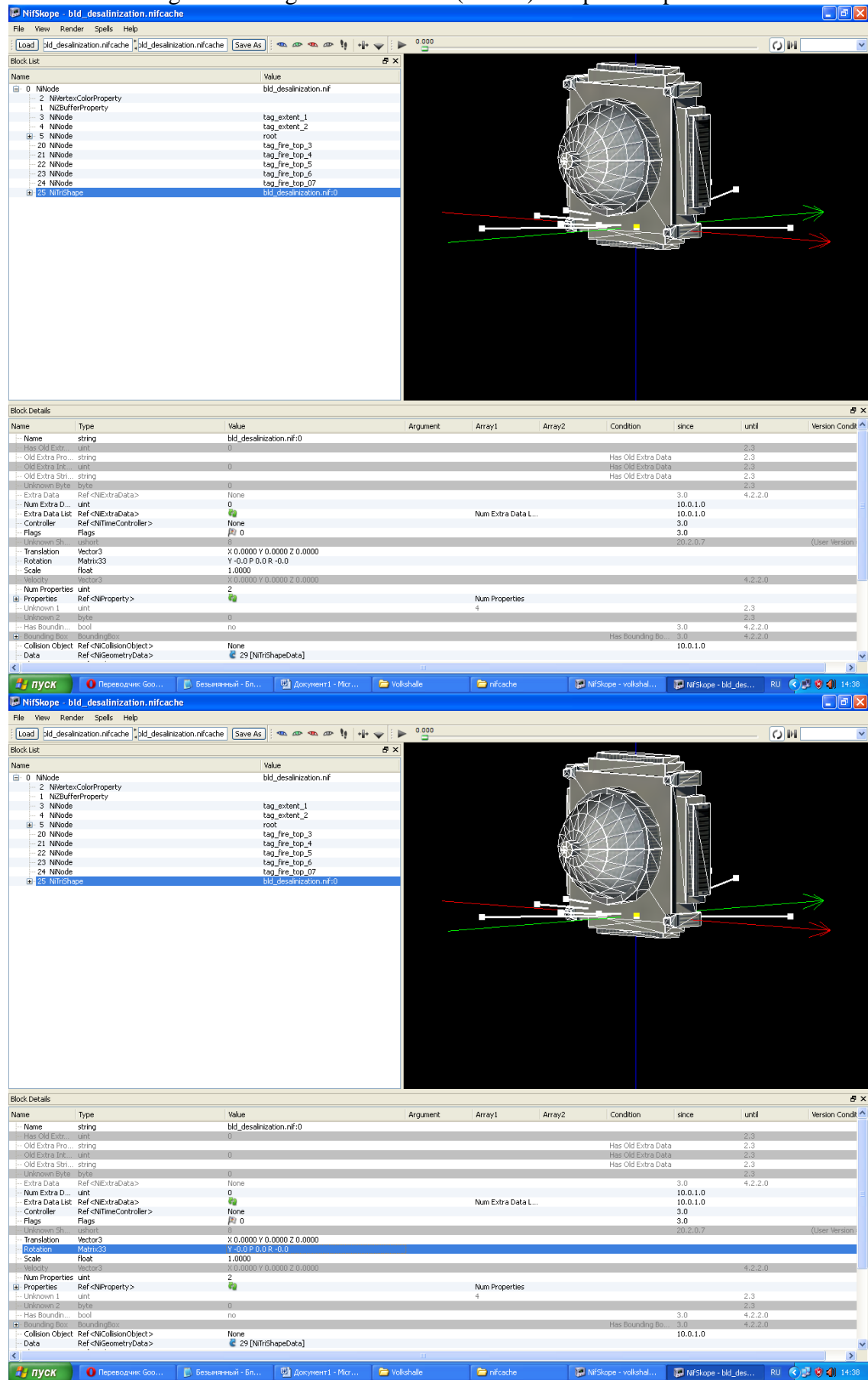
Select file 11.obj

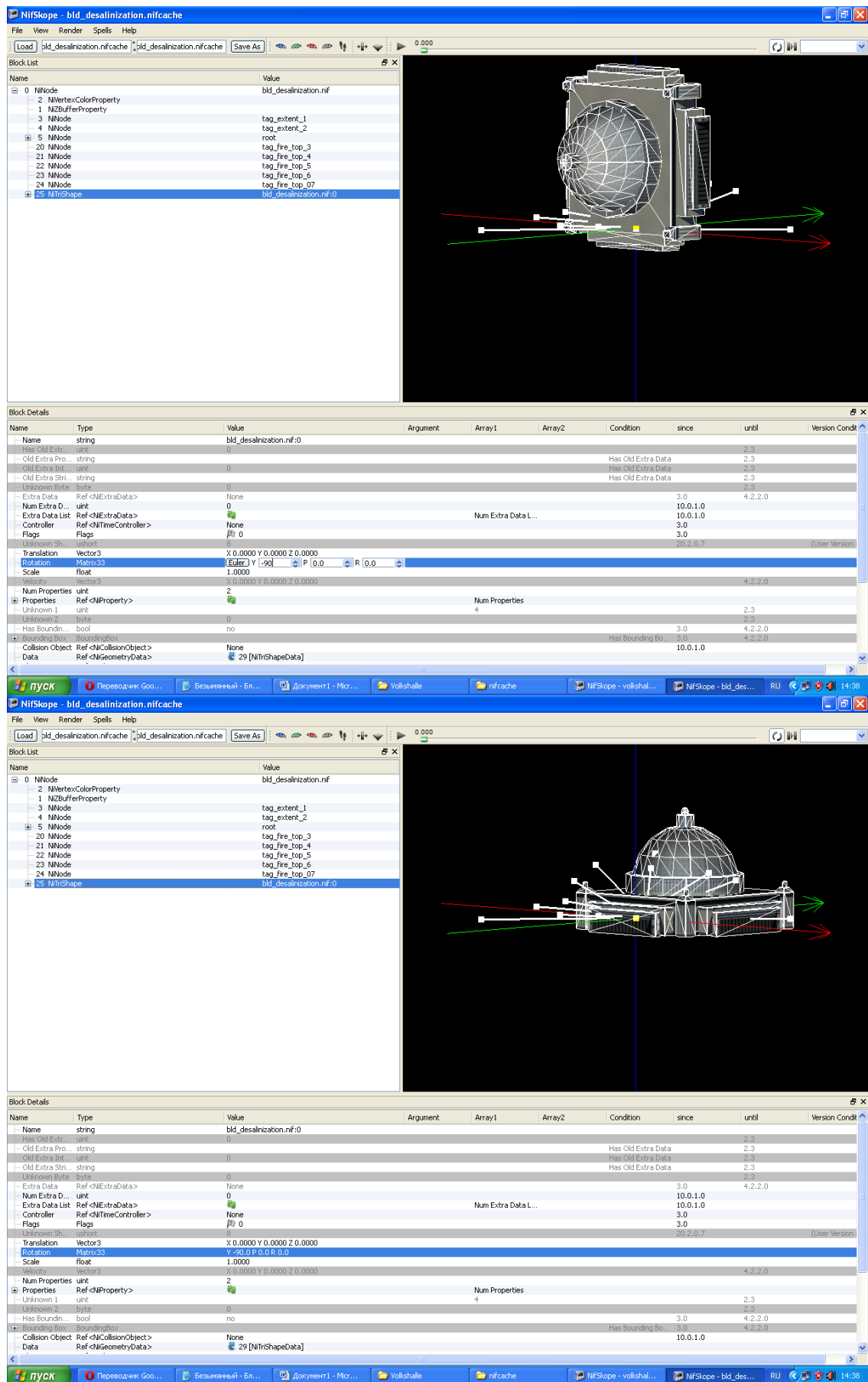
We receive:

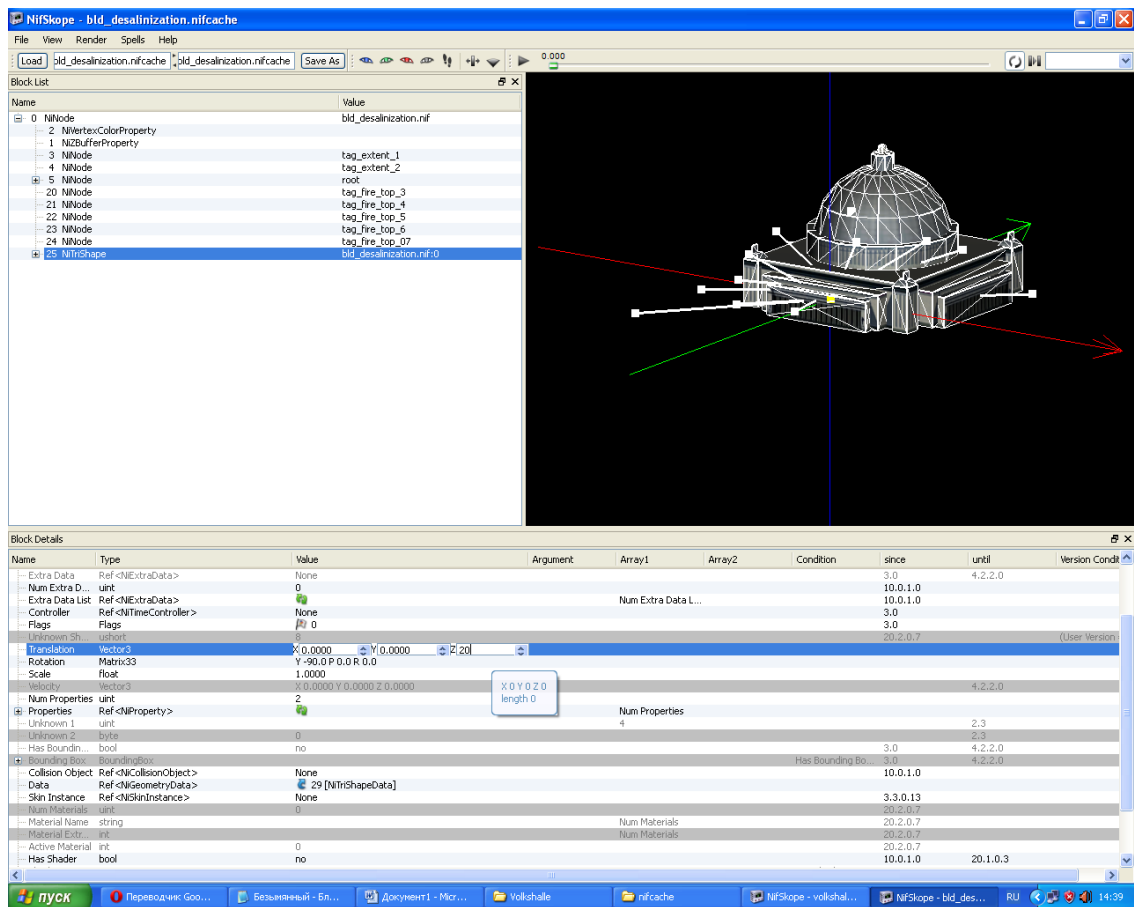
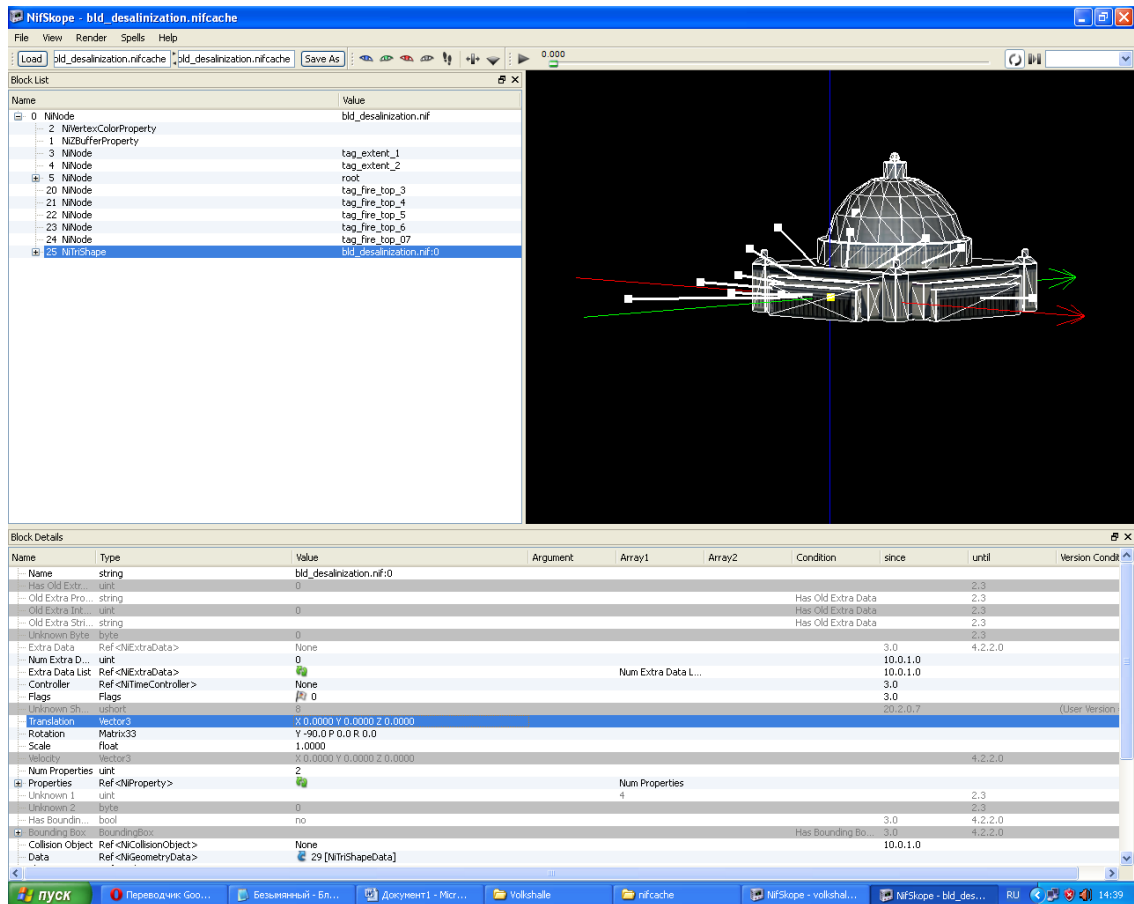


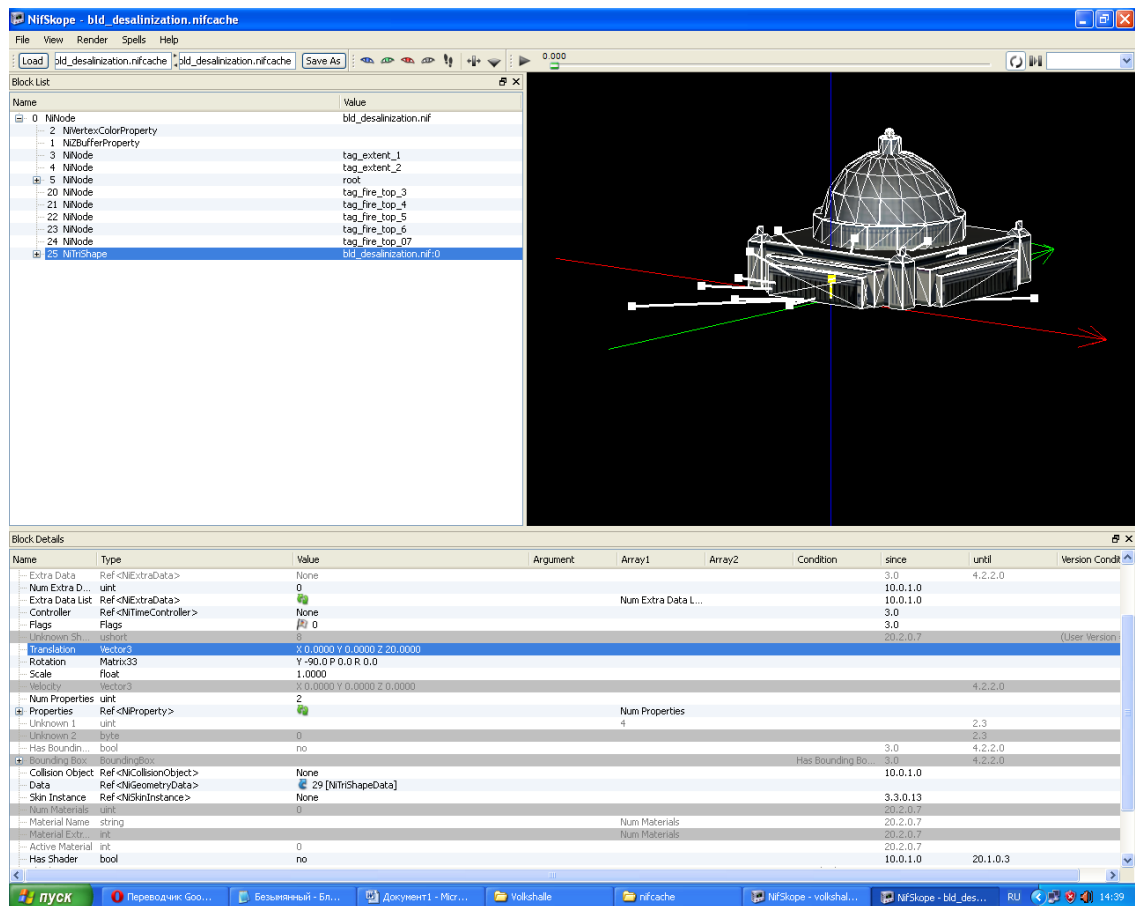


The new building is at an angle. Let's rotate it(Rotate) and pick it up.

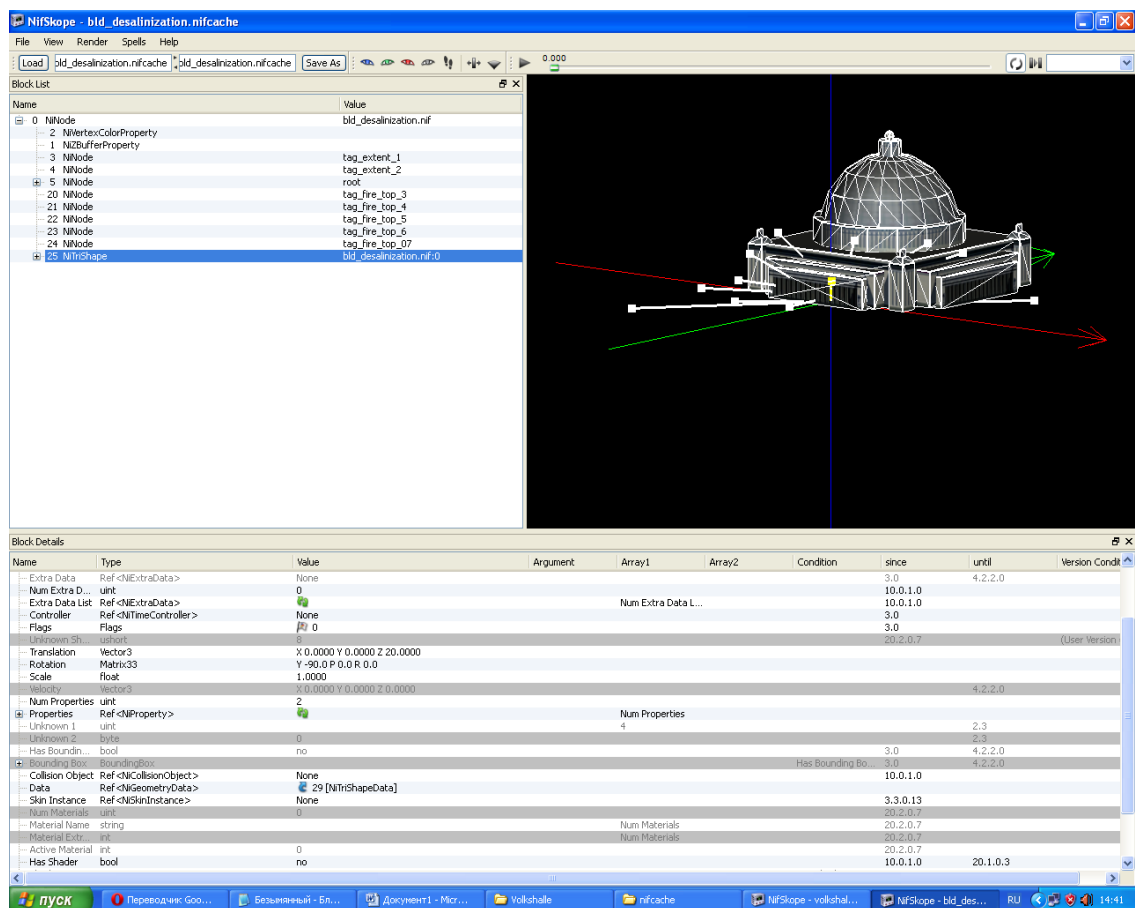


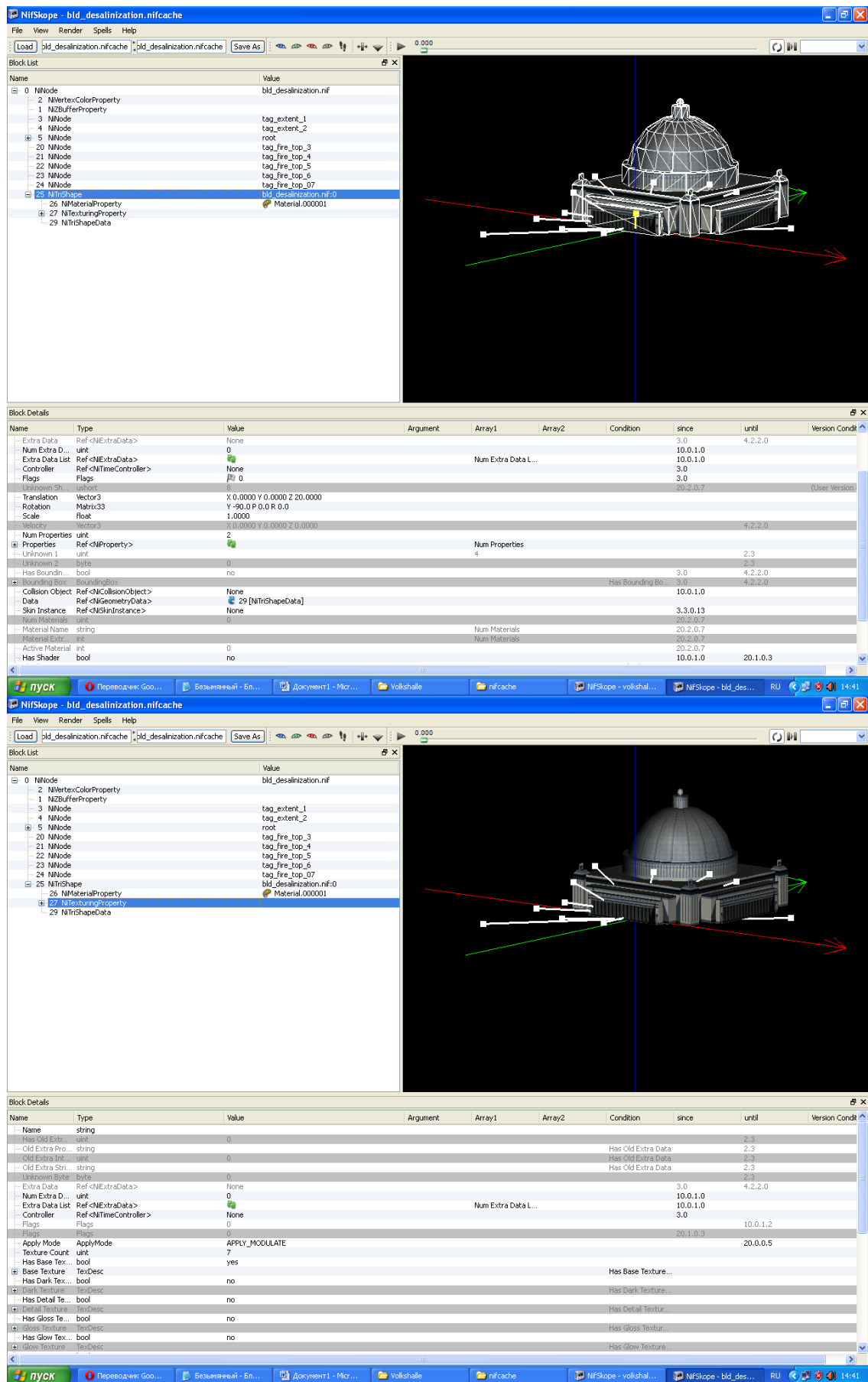


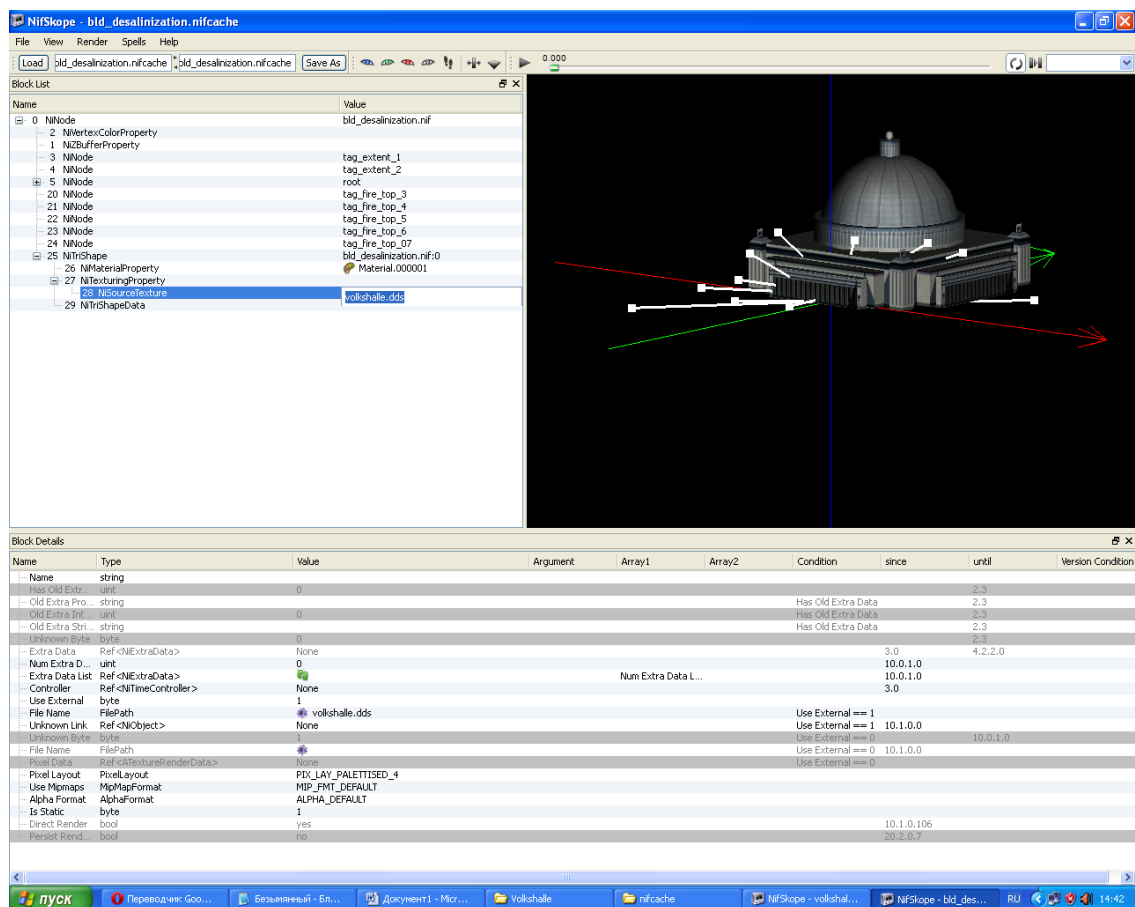
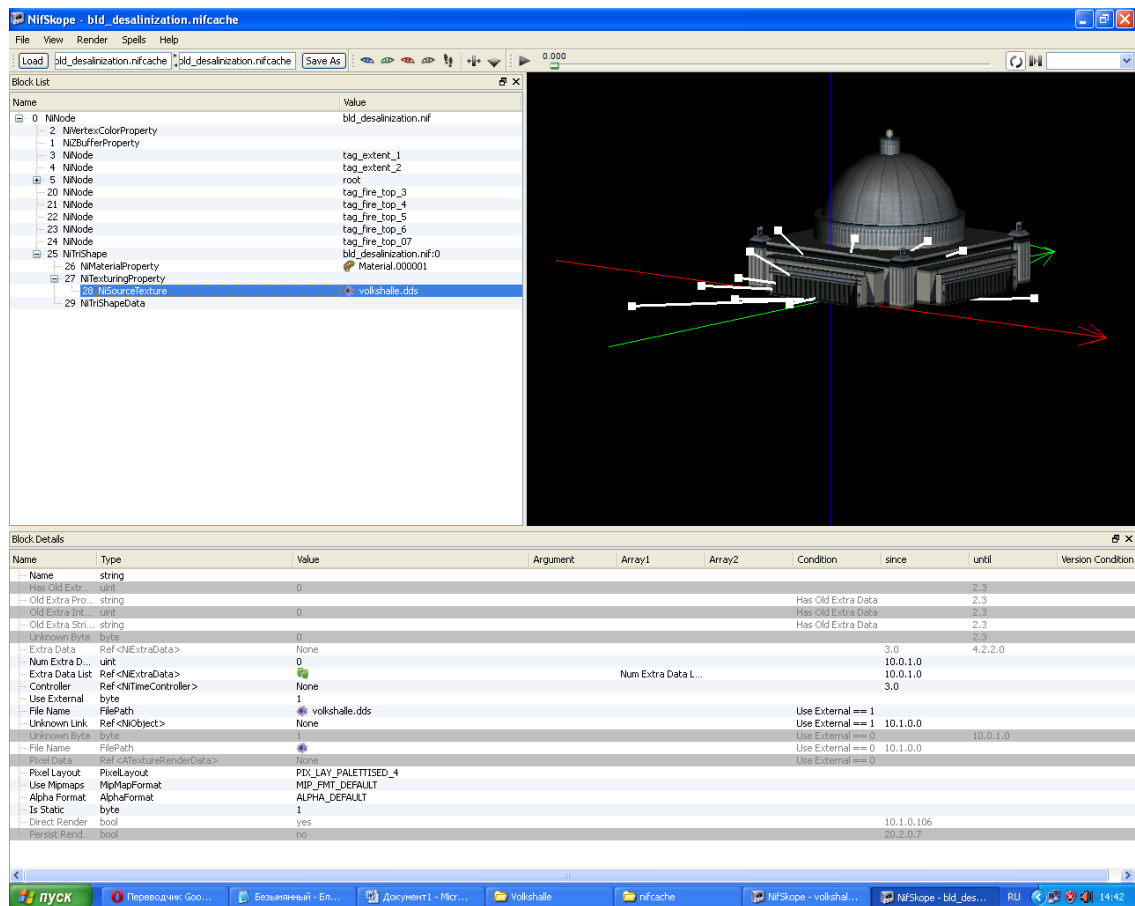




Now let's add the texture. (Texture must be in tga format)

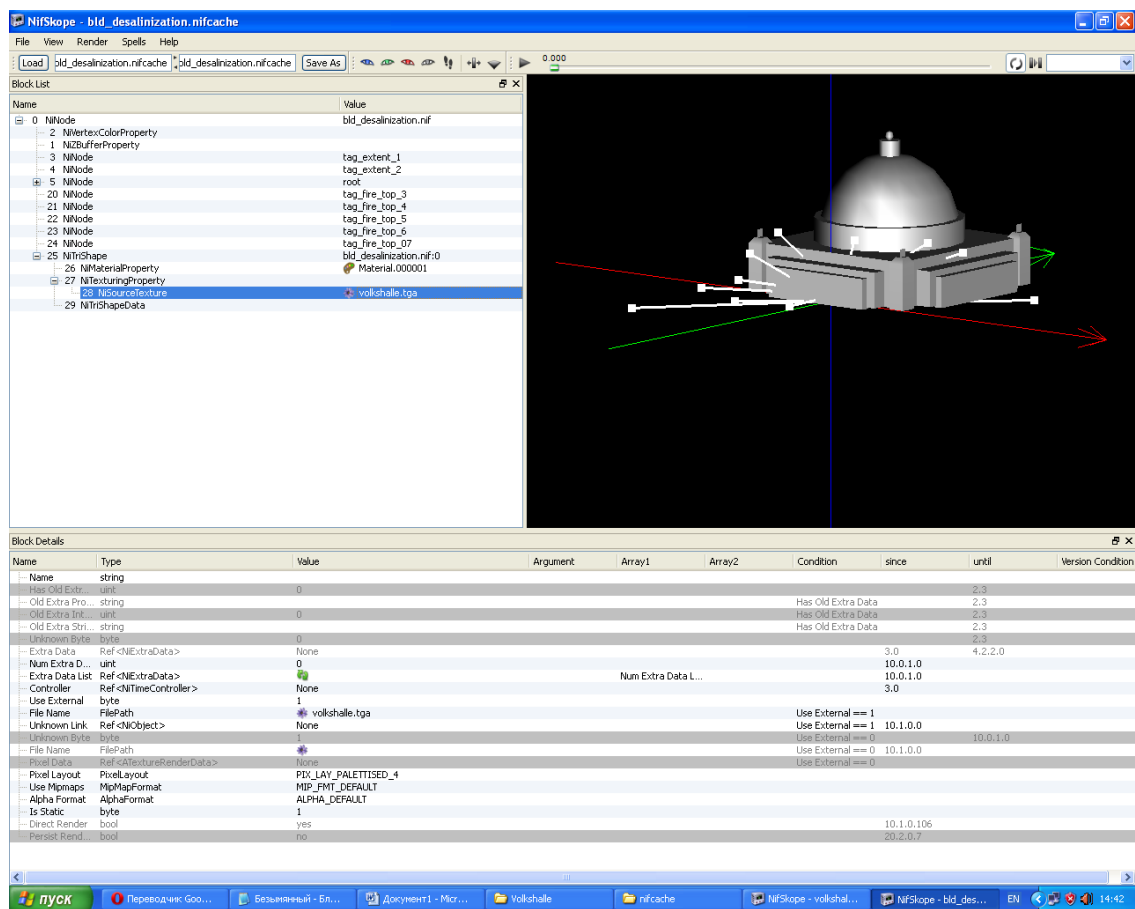
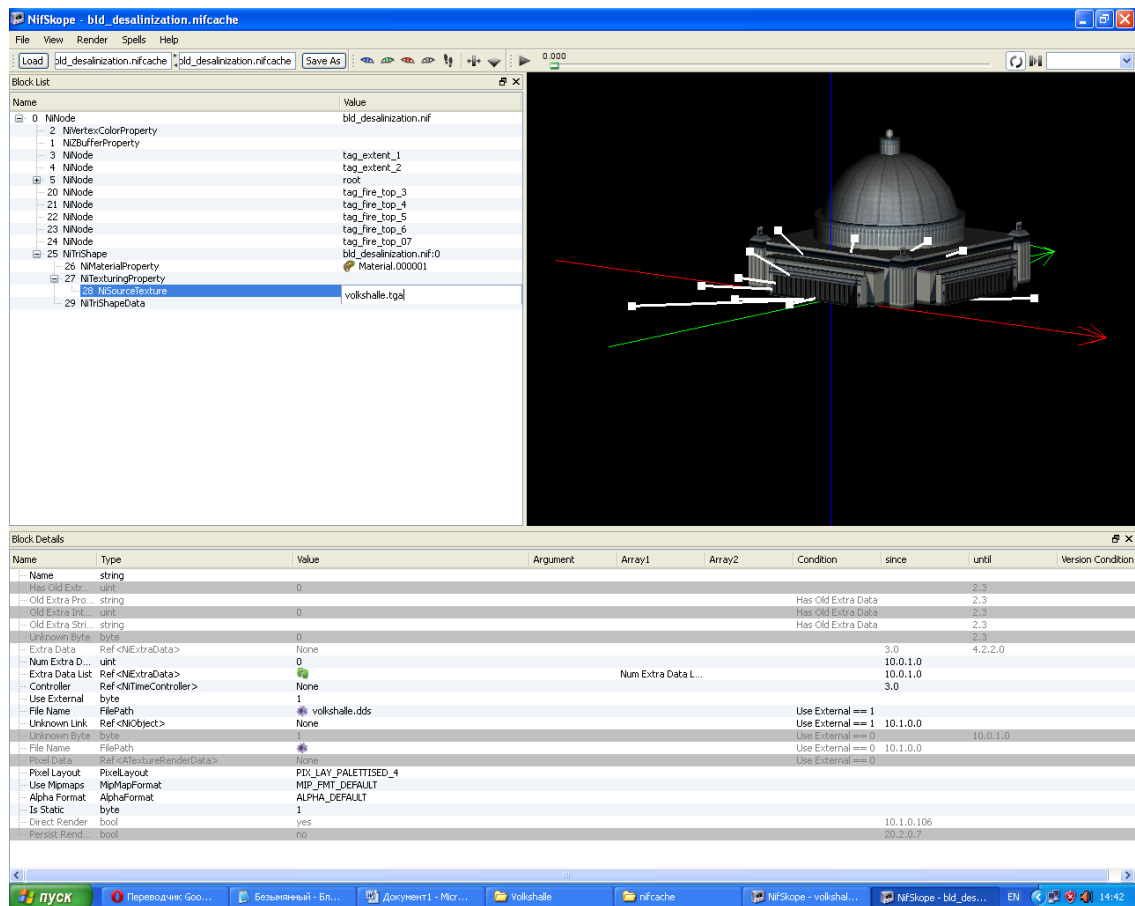




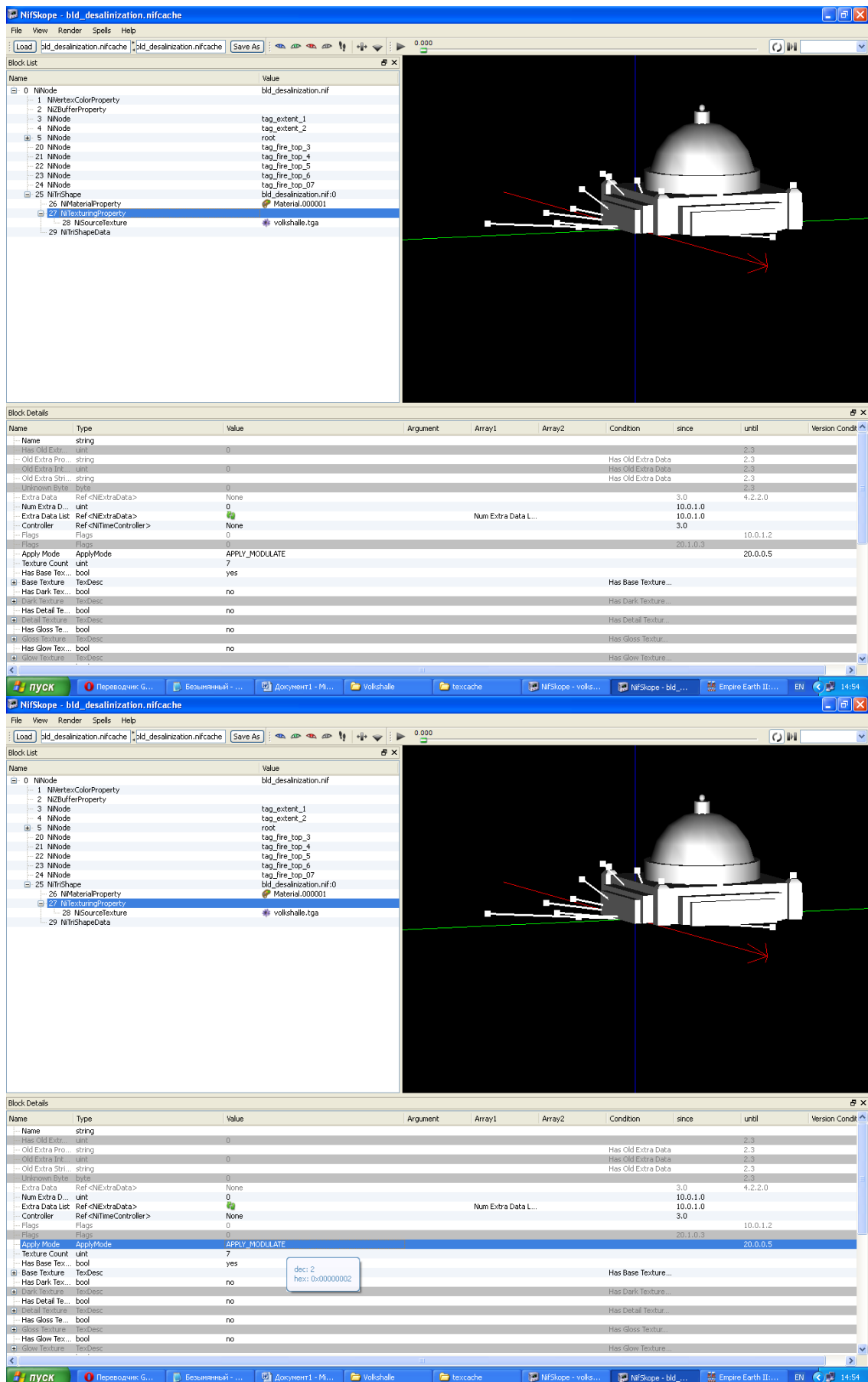


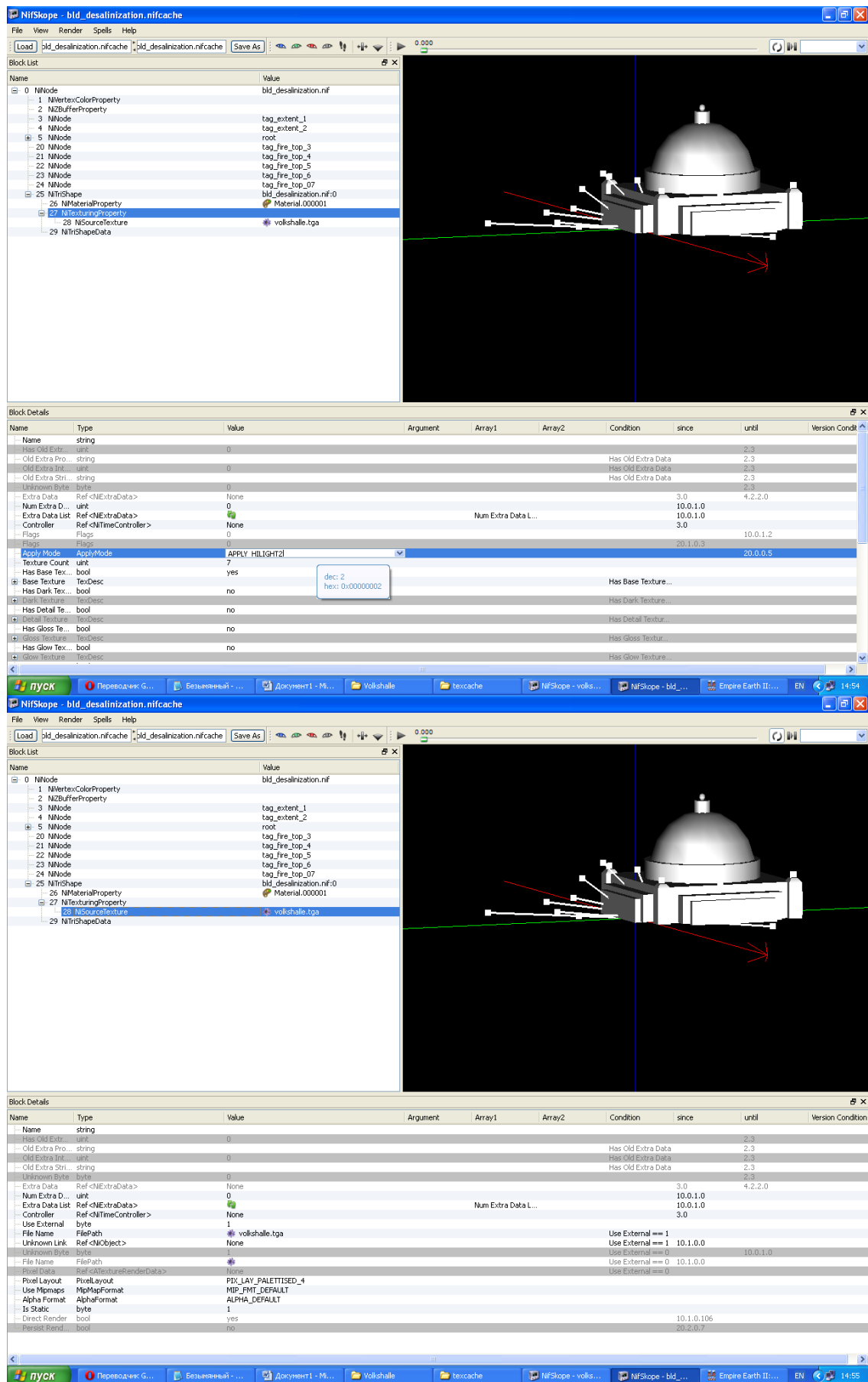
Let's write the name of the texture we need

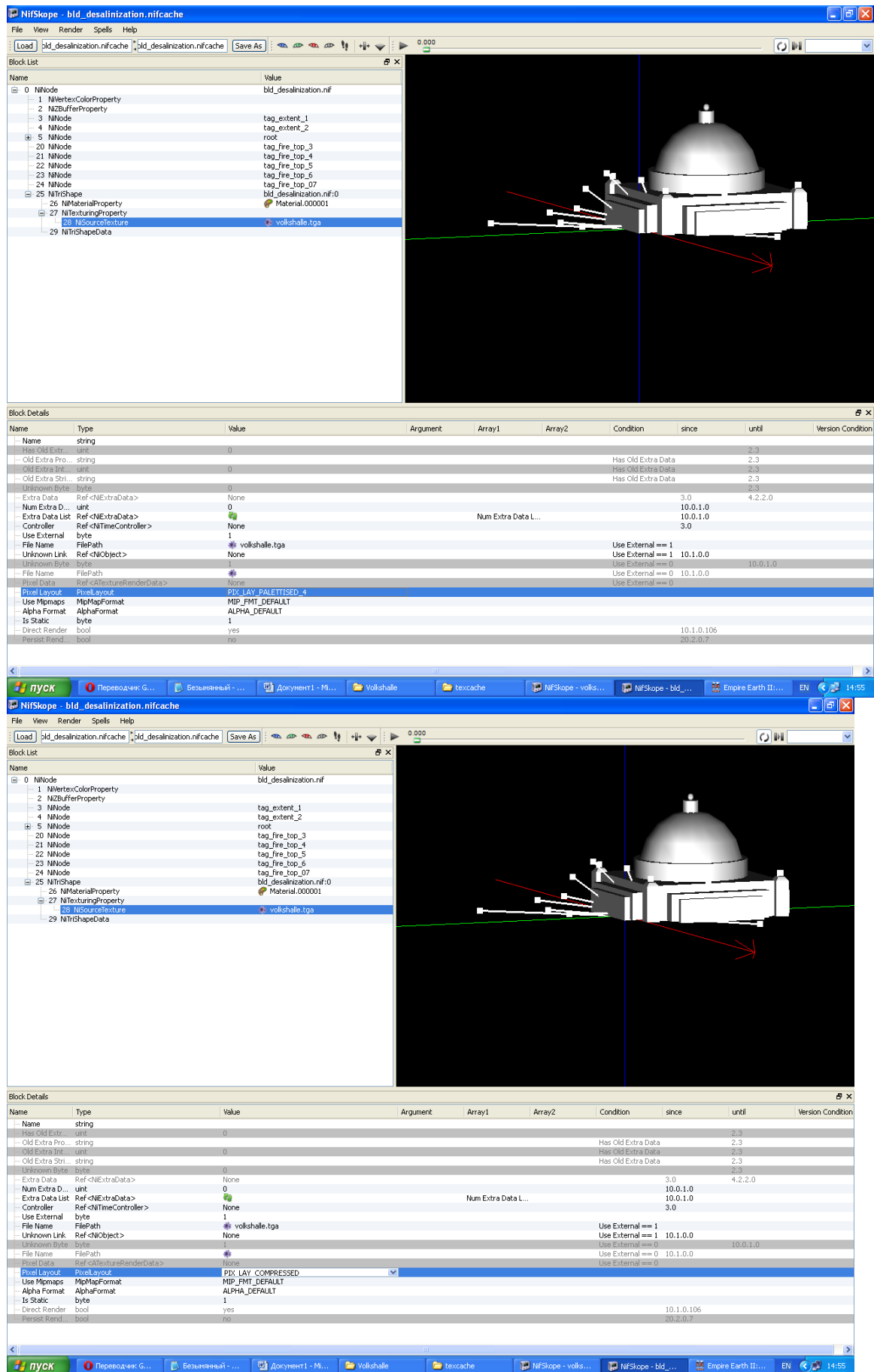


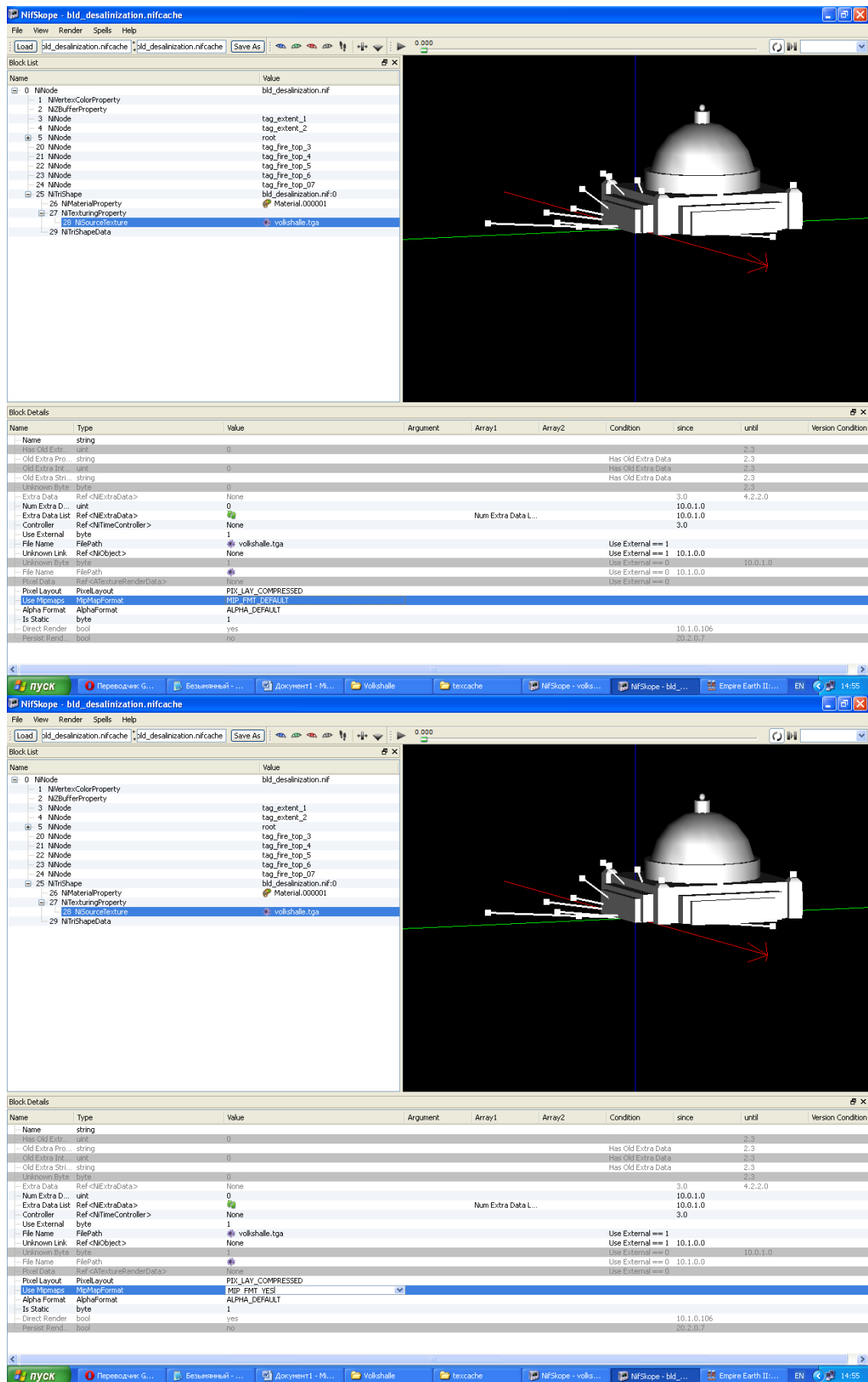


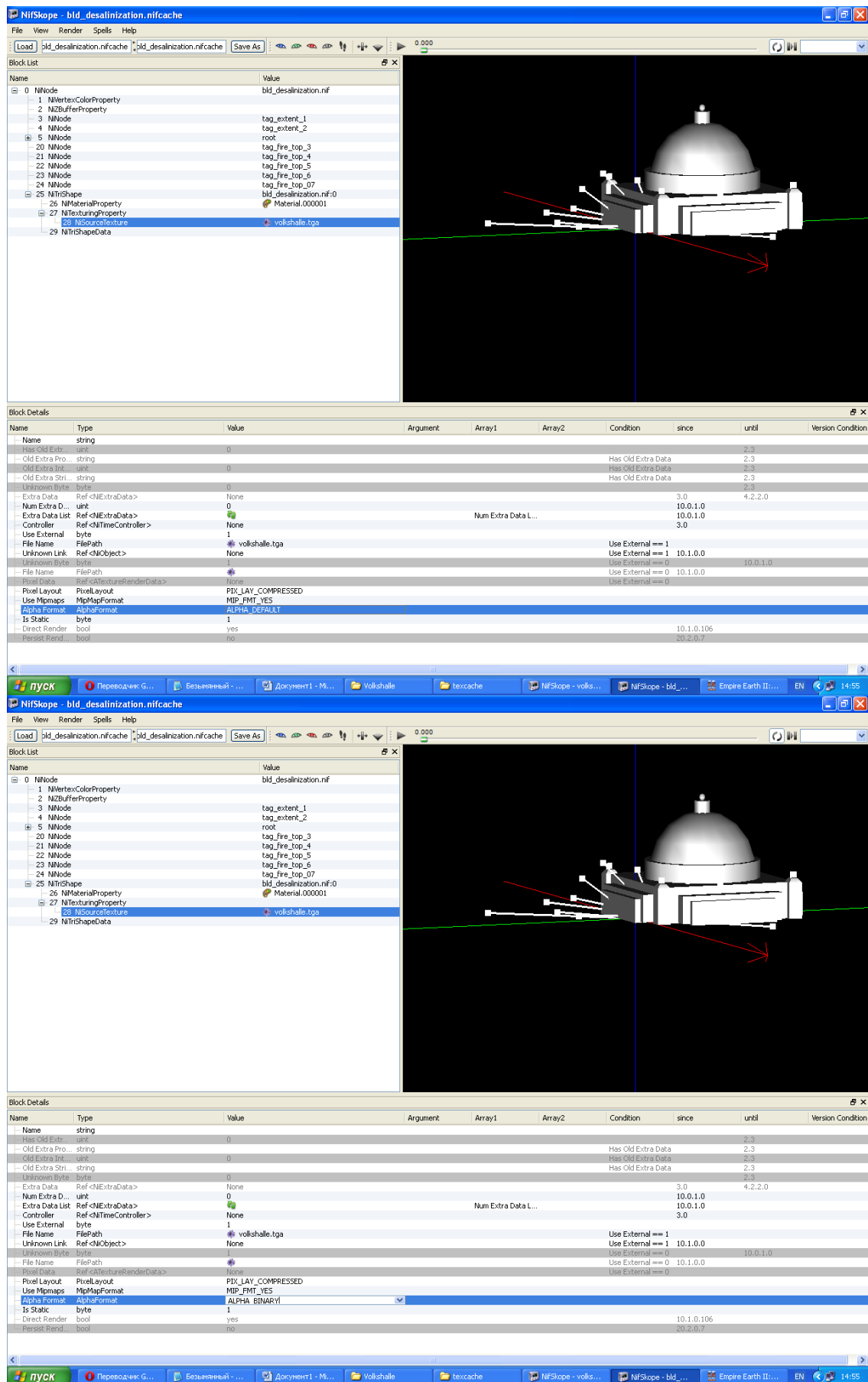
Now let's lighten the texture (this is necessary for some graphics cards, usually older versions)





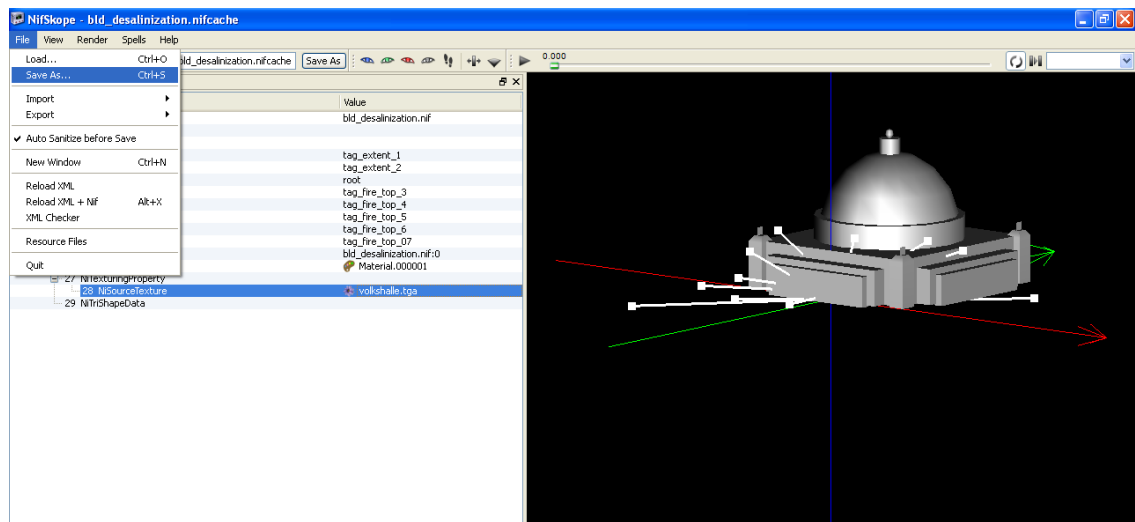




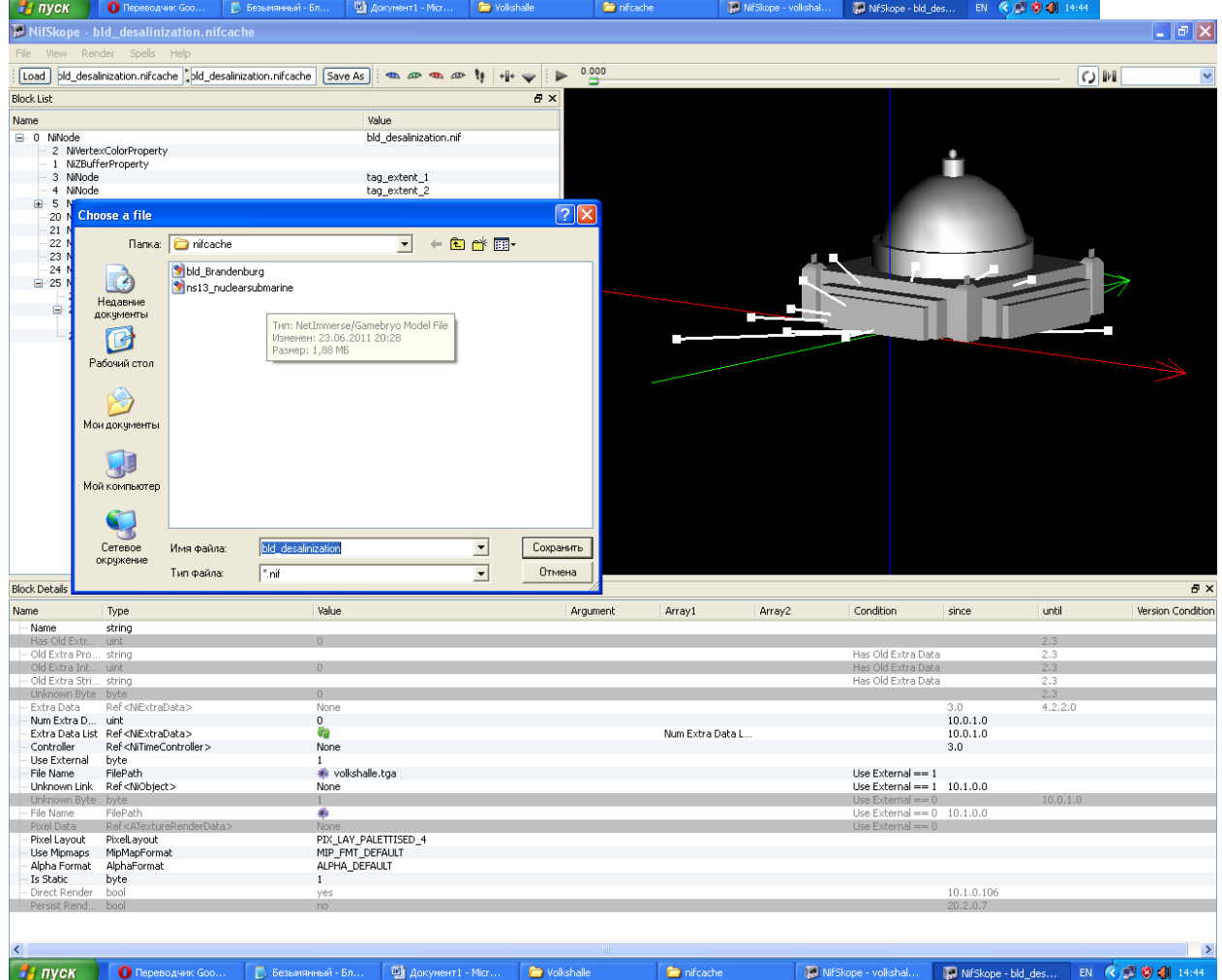


Save the model

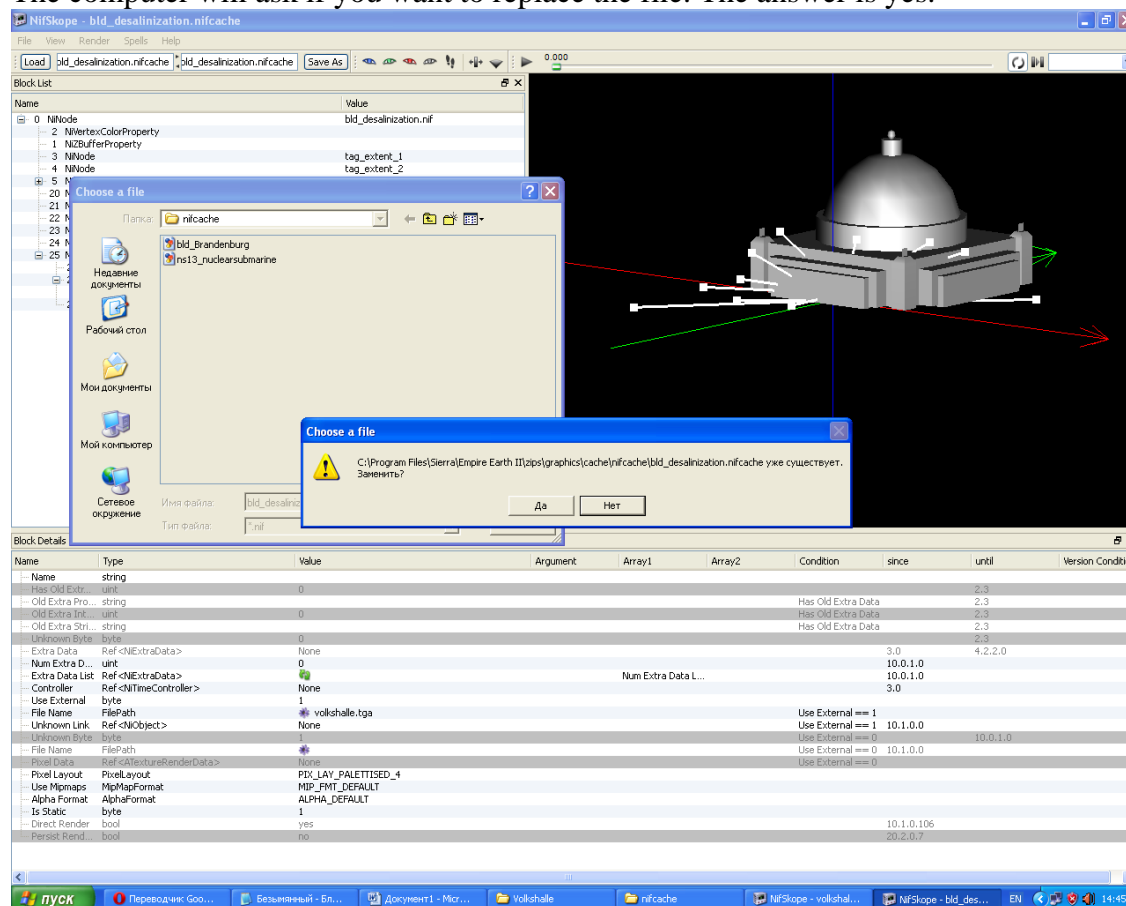




Name	Type	Value	Argument	Array1	Array2	Condition	since	until	Version Condition
Name	string								
Has Old Extra...	uint	0						2.3	
Old Extra Pro...	string					Has Old Extra Data		2.3	
Old Extra Int...	uint	0				Has Old Extra Data		2.3	
Old Extra Stri...	string					Has Old Extra Data		2.3	
Unknown Byte...	byte	0						2.3	
Extra Data	Ref<NExtraData>	None					3.0	4.2.2.0	
Num Extra D...	uint	0					10.0.1.0		
Extra Data List	Ref<NExtraData>						10.0.1.0		
Controller	Ref<NTimeController>	None			Num Extra Data L...		3.0		
Use External	byte	1							
File Name	FilePath	volkshalle.tga				Use External == 1			
Unknown Link	Ref<NIOObject>	None				Use External == 1	10.1.0.0		
Unknown Byte...	byte	1				Use External == 0		10.0.1.0	
File Name	FilePath					Use External == 0		10.1.0.0	
Pixel Data	Ref<TextureRenderData>					Use External == 0			
Pixel Layout	PixelFormat	PIX_FMT_PALETTE4							
Use Mipmaps	MipMapFormat	MIP_FMT_DEFAULT							
Alpha Format	AlphaFormat	ALPHA_DEFAULT							
Is Static	byte	1							
Direct Render	bool	yes					10.1.0.106		
Persist Rend...	bool	no					20.2.0.7		



The computer will ask if you want to replace the file. The answer is yes.



Now let's open the game folder again:

C: \ Program Files \ Sierra \ Empire Earth II

Find the archive in that folder named textures.zip

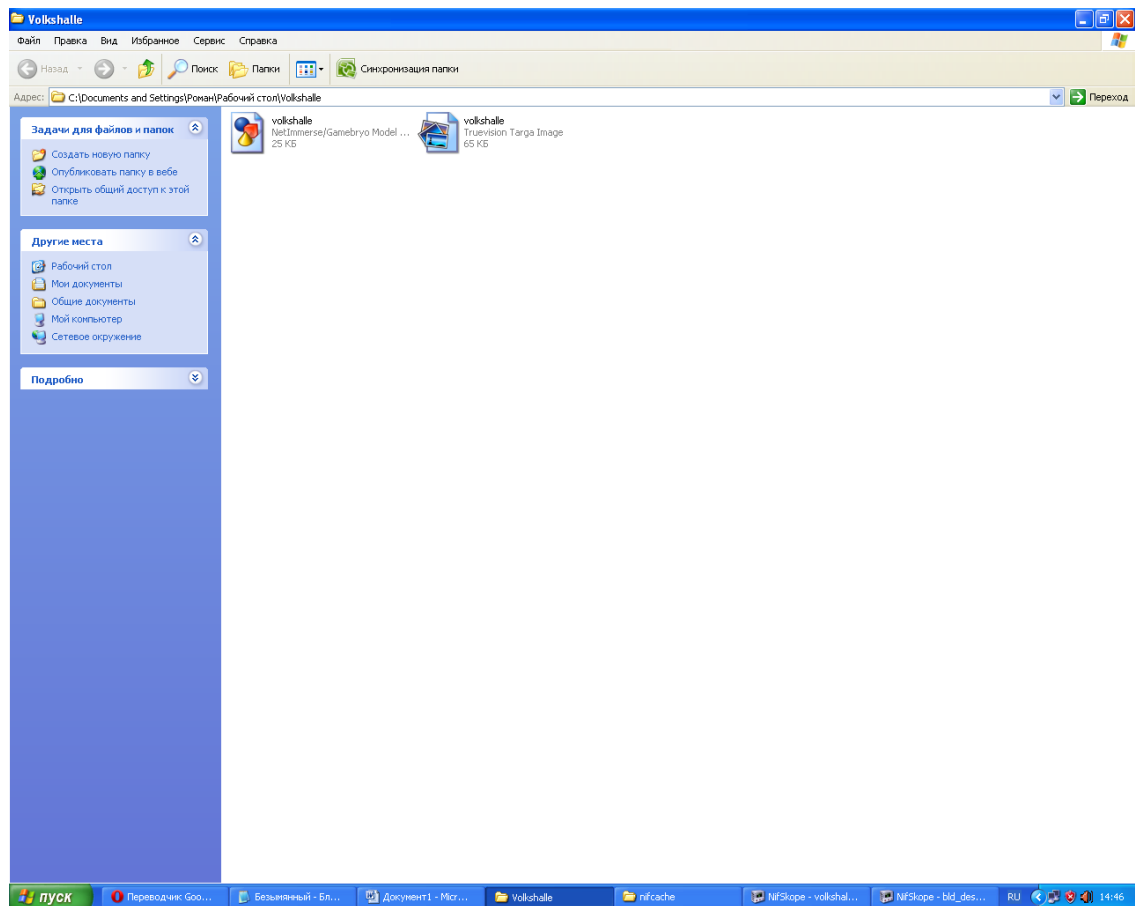
C: \ Program Files \ Sierra \ Empire Earth II \ zips \ textures.zip

Unpack this archive, you should get a textures folder. Now the archive can be deleted or moved elsewhere (for example to the desktop) Then open the texcache folder located at:

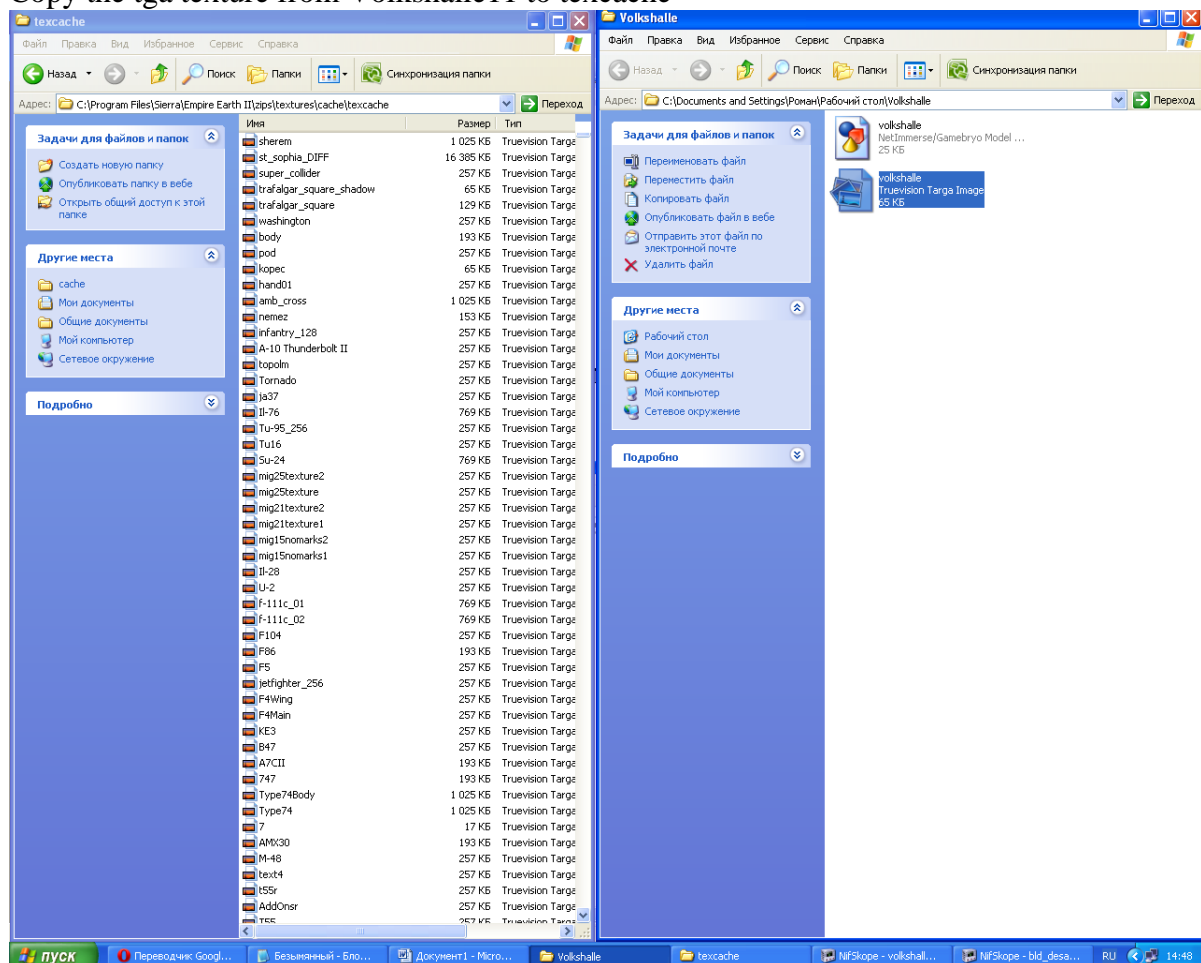
(C: \ Program Files \ Sierra \ Empire Earth II \ zips \ textures \ cache \ texcache)

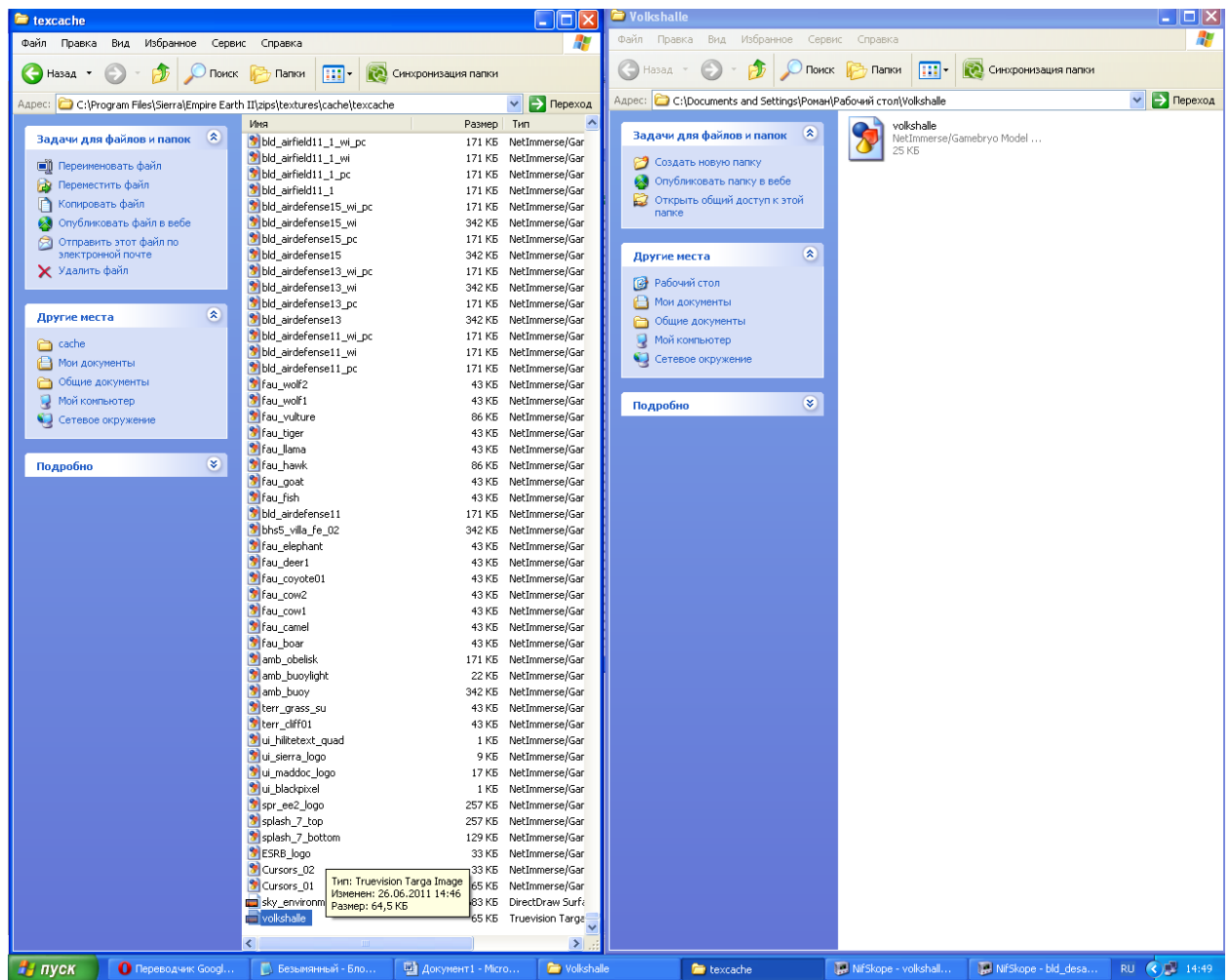
This folder contains all the textures in the game.

In parallel, open the folder Volkshalle11



Copy the tga texture from Volkshalle11 to texcache





Everything is ready!!!

Launch Empire Earth 2  
Open map editor  
Choose a desalination plant.  
Let's see what we have:



P.S.

With this technique, everything is done the same way. The easiest way is to customize your jets and submarines, delete the old model then import the new one. If you want to make a tank, you should have at least two parts in your adaptable model: the first is the turret with the gun, the second is the hull of the tank with the chassis. Replace every original part of the tank with a new one: the old turret with a new one, the old hull with a new one - this is the main principle of operation. It all depends on your experience, work better with simpler models initially.

## 7 How to add a new 3D human or animal model to the game.

( !!! Important: Read this chapter at least once before starting work. It is very desirable to have skills in working with a blender program !!!)

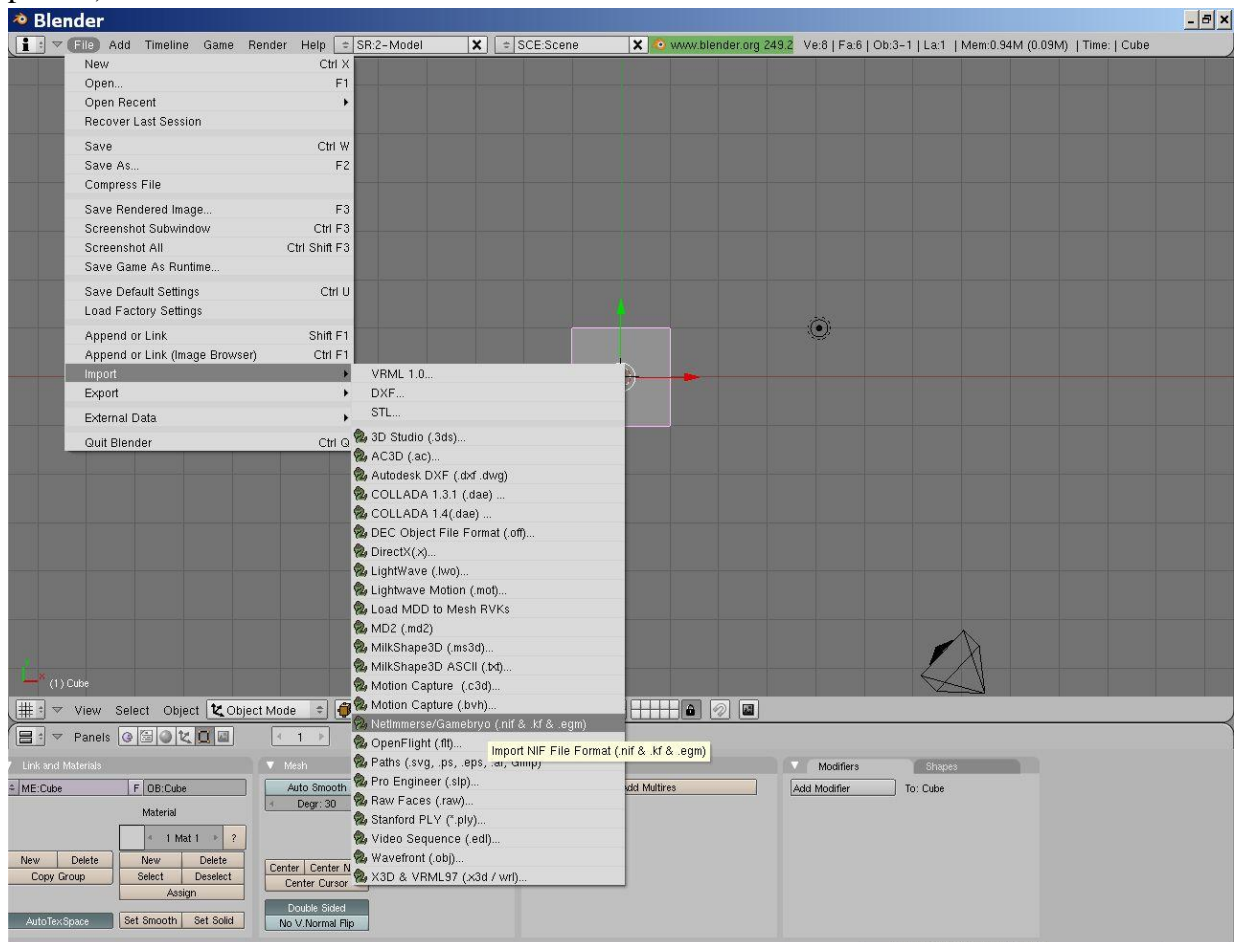
Author P.M.A.

We need the following programs:

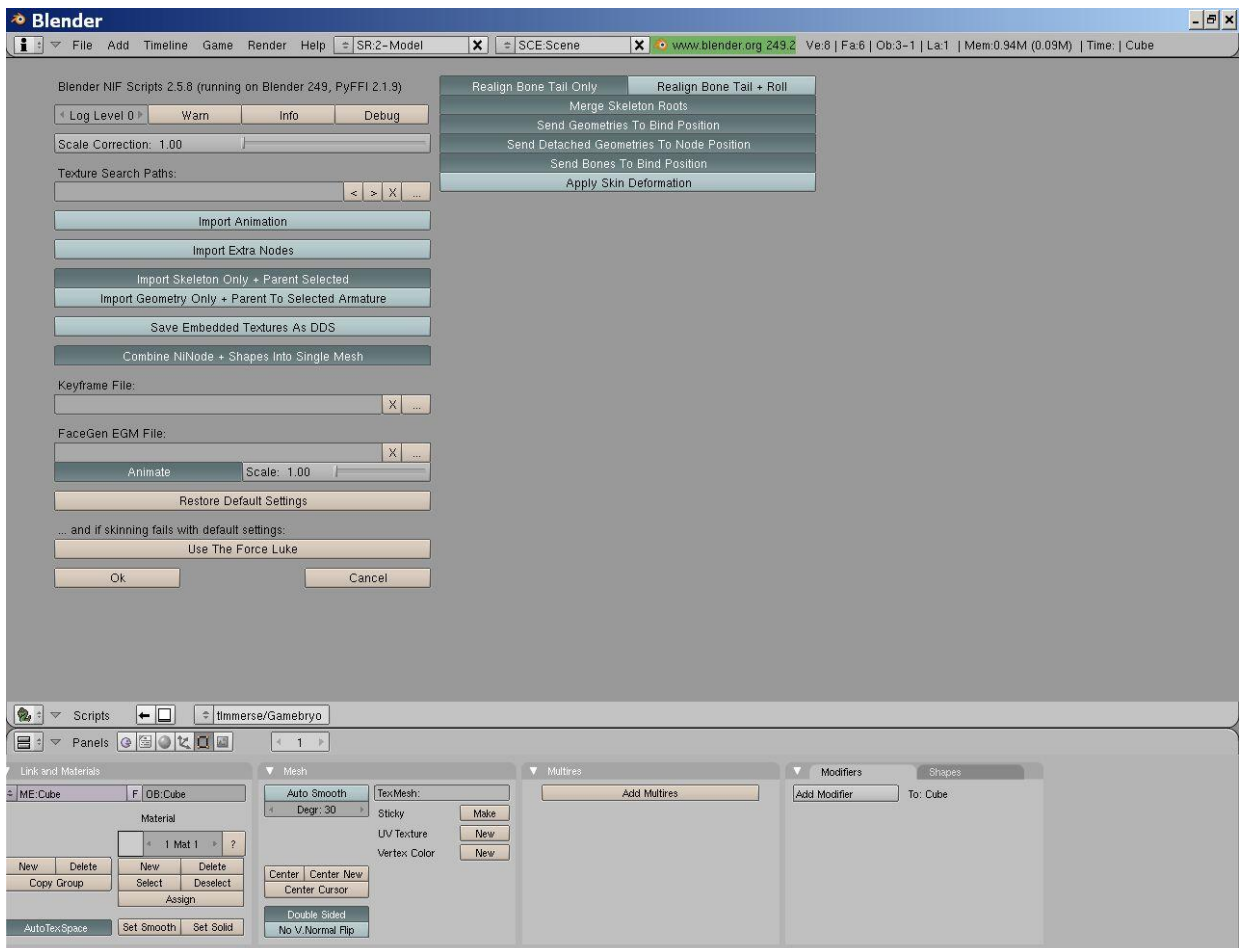
- Python 2.6.6
- PyFFI 2.1.9
- Blender 2.49b
- Blender NIF Scripts 2.5.07
- NifSkopec

Download the first 4 from this link: <http://niftools.sourceforge.net/wiki/Blender>  
We install them in the same order as presented above.

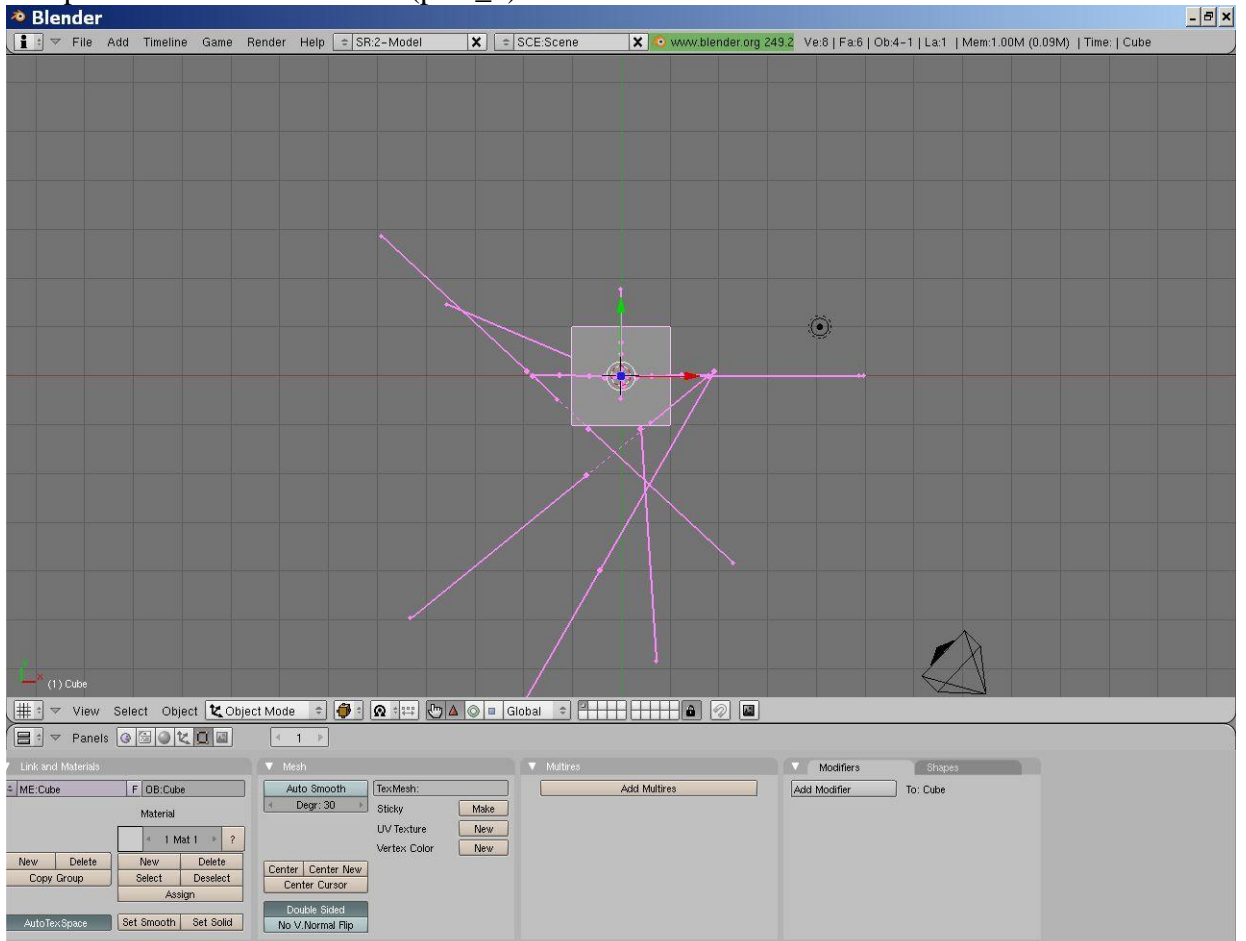
Run Blender 2.49b and import the model skeleton `lh11_rifleman_we.nifcache` (`pic1_1` and `pic1_2`).



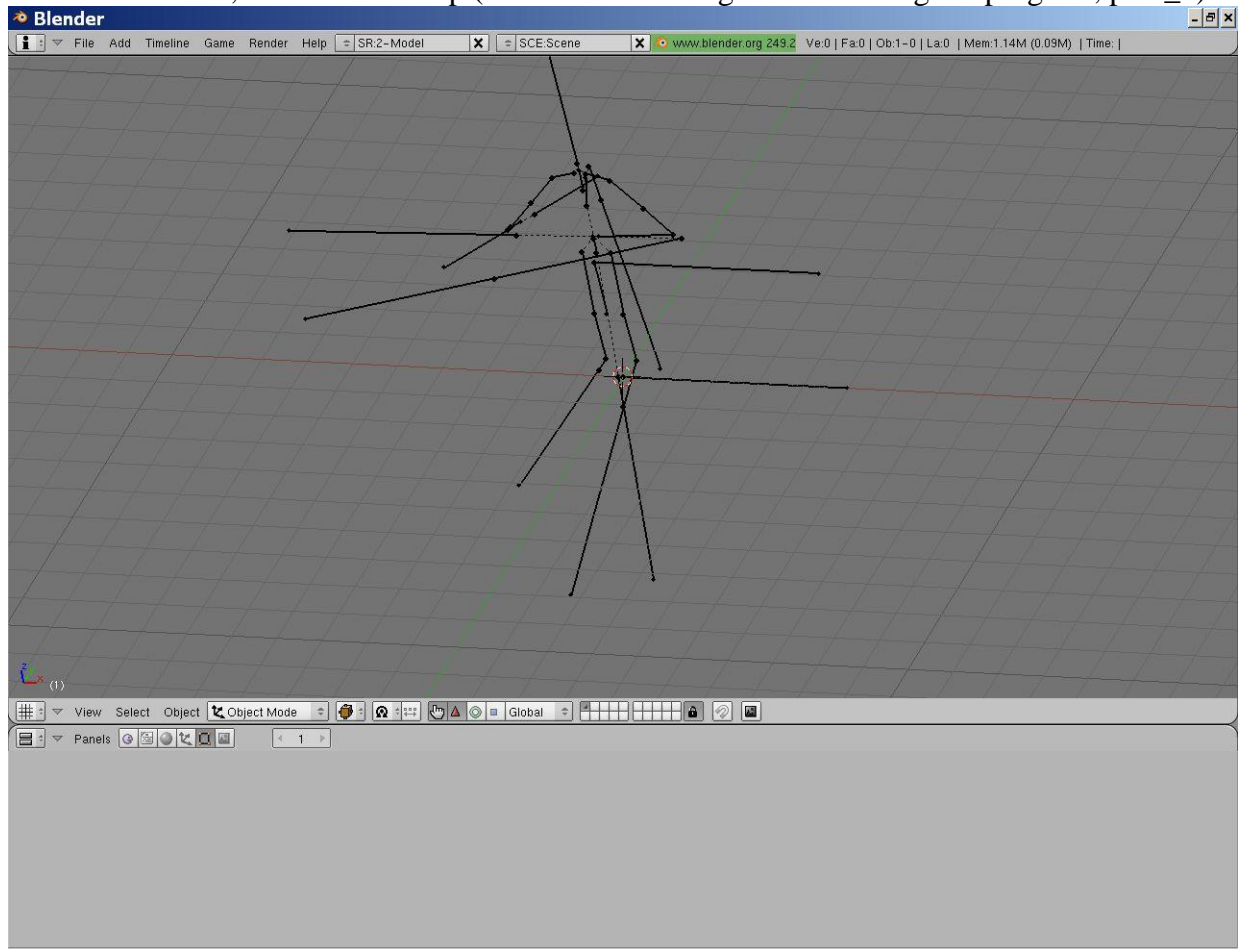




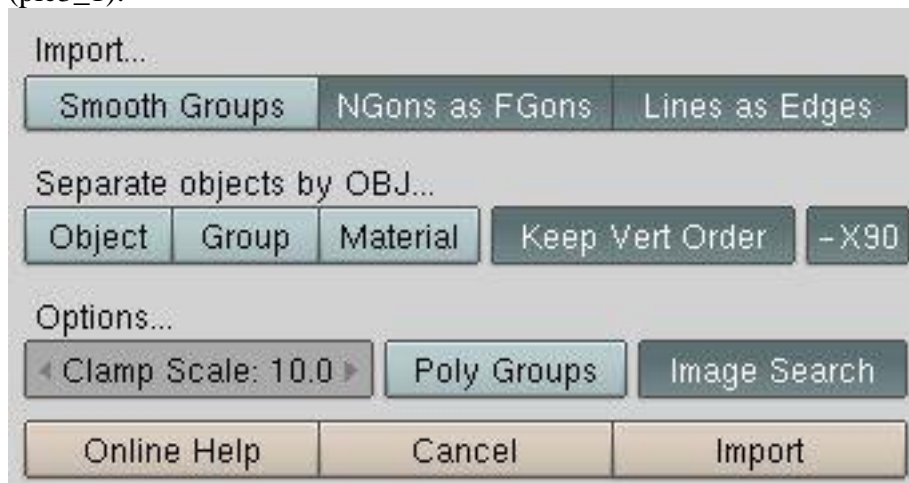
The pink lines are our skeleton (pic2\_1).



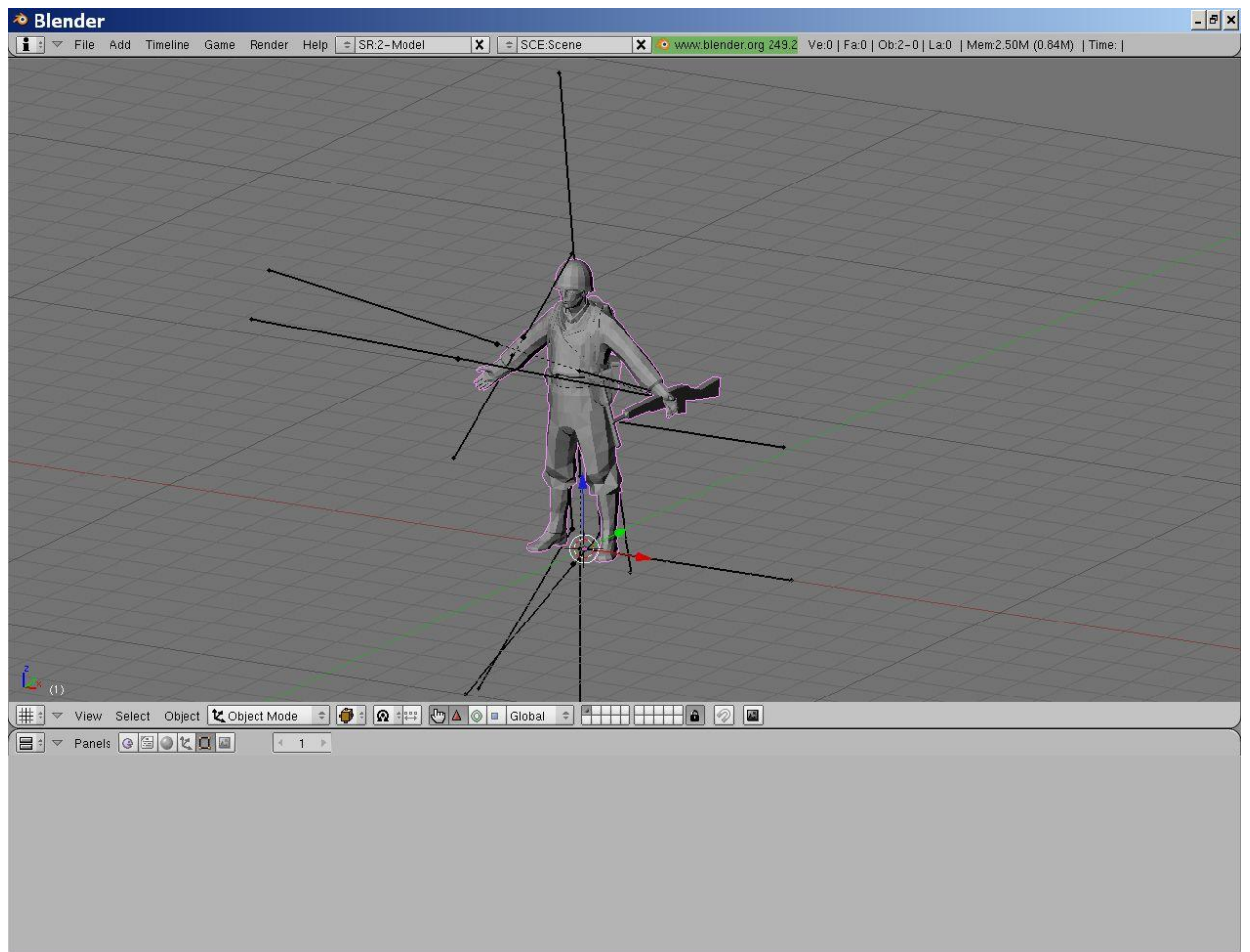
Remove the cube, camera and lamp (this can be done right after starting the program, pic2\_2).



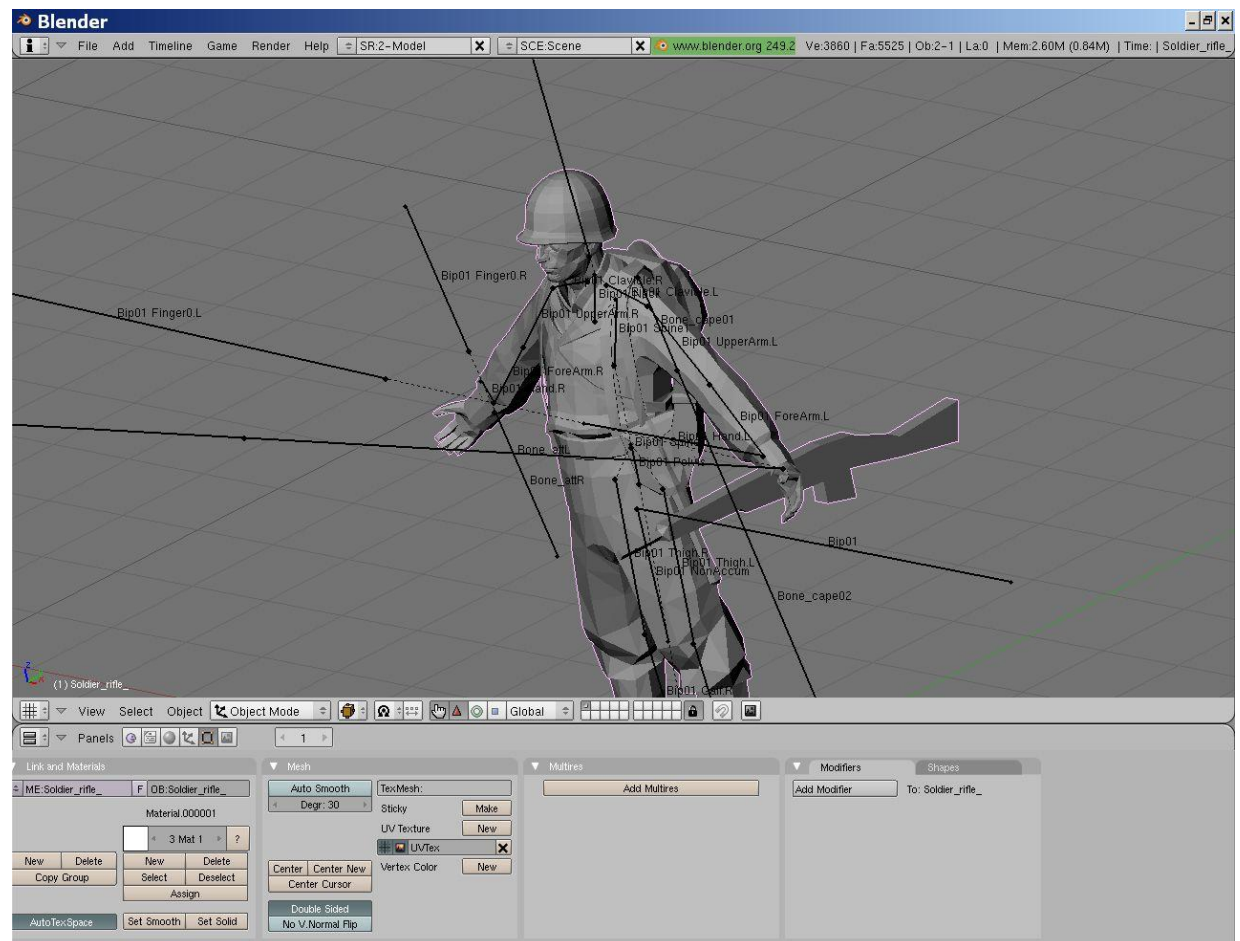
Now let's import our model. I originally had it in obj. format. I liked these import settings (pic3\_1):



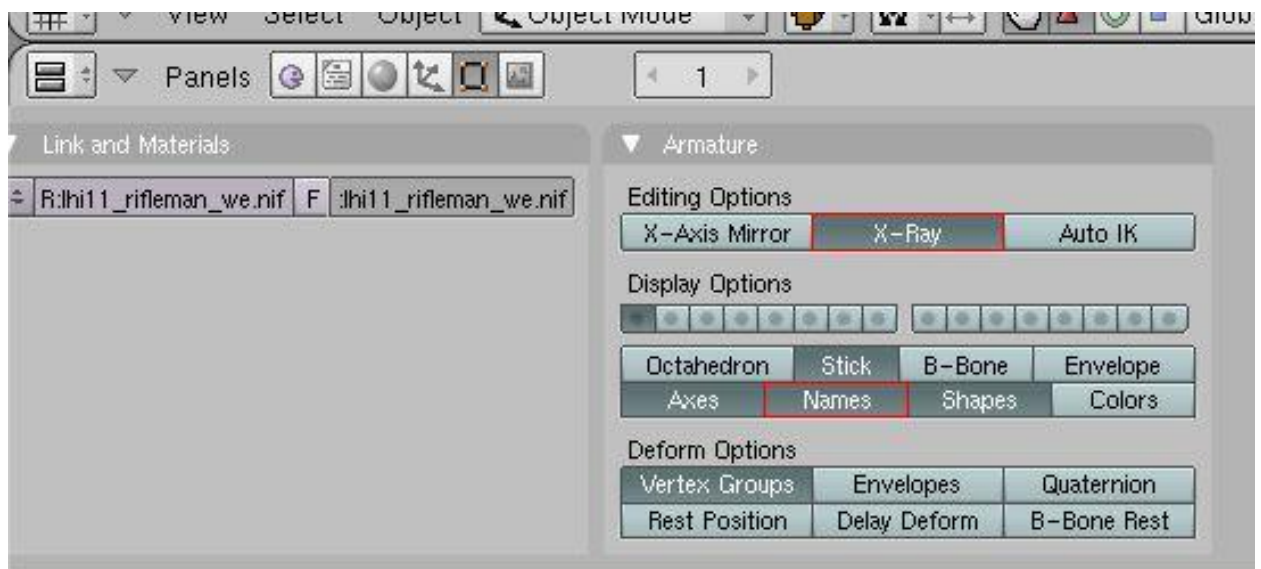
After importing, the scene looks like this(pic3\_2):



Now we adjust our model to the size and position of the skeleton (pic4\_1).

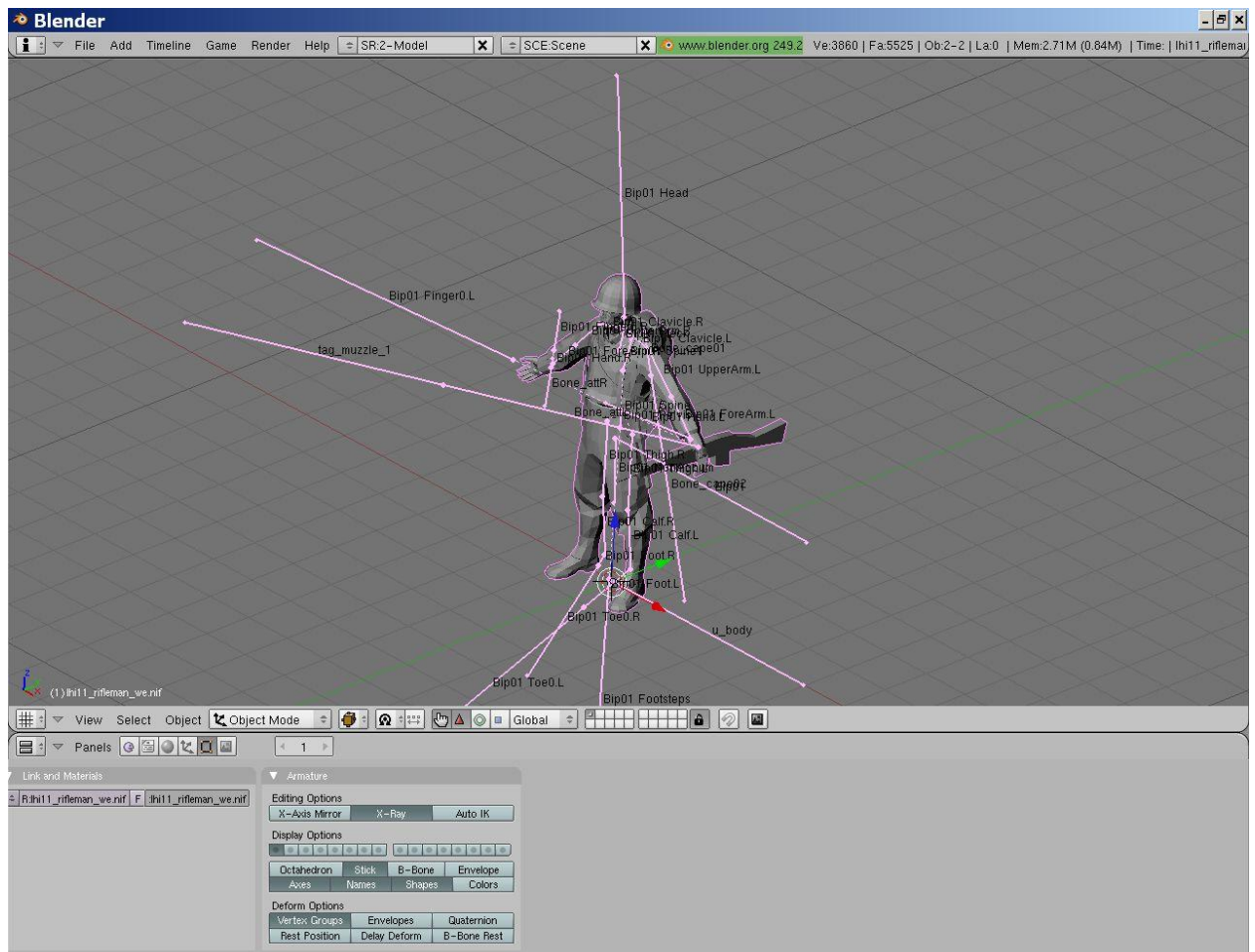


For convenience, I chose the skeleton and turned on the display of bone names and the X-Ray mode, which shows bones hidden behind the model (pic4\_2).



Then select the model and the skeleton (strictly in that order - model first, then skeleton). To do this, hold down Shift and right-click select the model, then the skeleton (pic5\_1).





If you have already checked, then click Ctrl + P and choose Armature (pic5\_2) and then Name Groups (pic5\_3).

### Make Parent To

Armature

Object

### Create Vertex Groups?

Don't Create Groups

Name Groups

Create From Envelopes

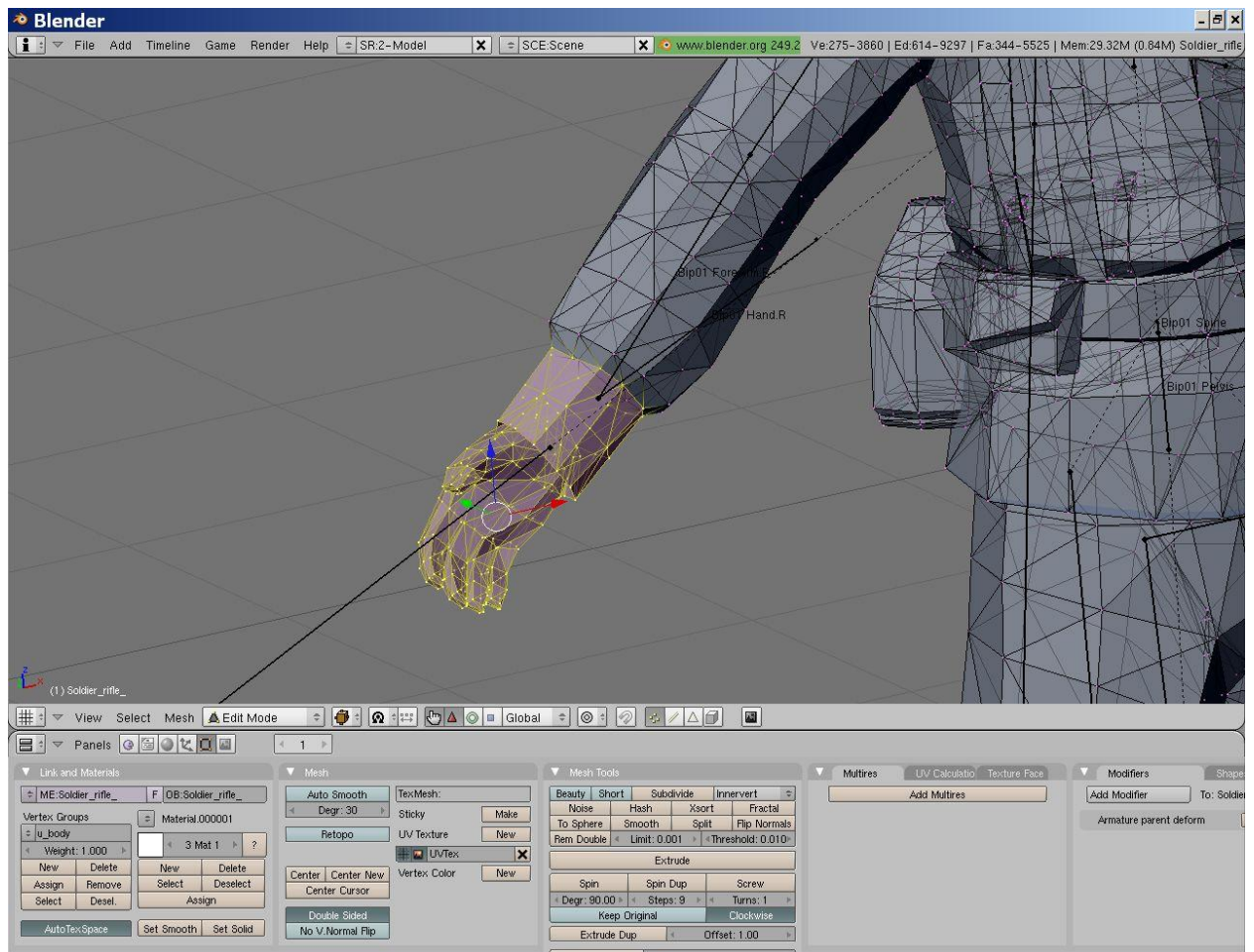
Create From Bone Heat

If you choose Create From Bone Heat, then the vertices will be automatically associated with nearby bones. This is how we created vertex groups with names like armature. And we will manually indicate which peaks and which group will be included.

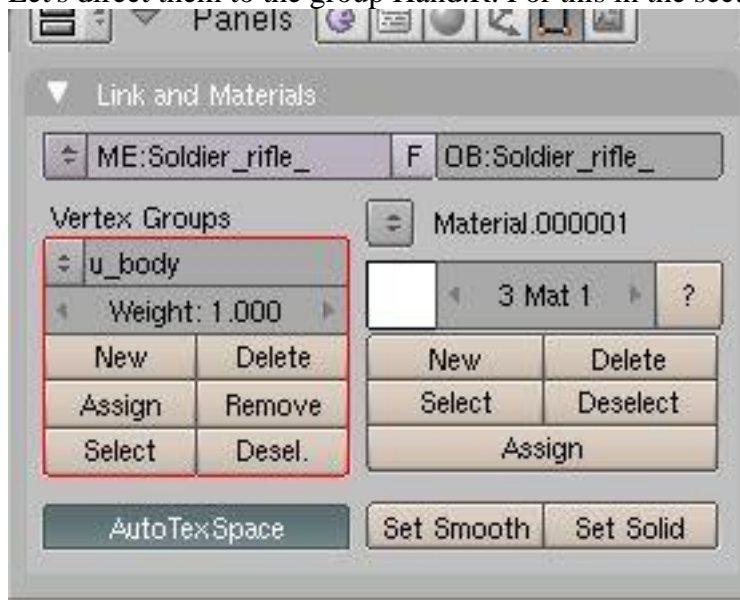
So we come to a long and tedious process - binding the vertices. Select the model and go to the edit mode (button Tab, pic6\_1). For convenience, you can disable / enable the face display mode (key Z) and turn off / on the texture display mode (Alt + Z).



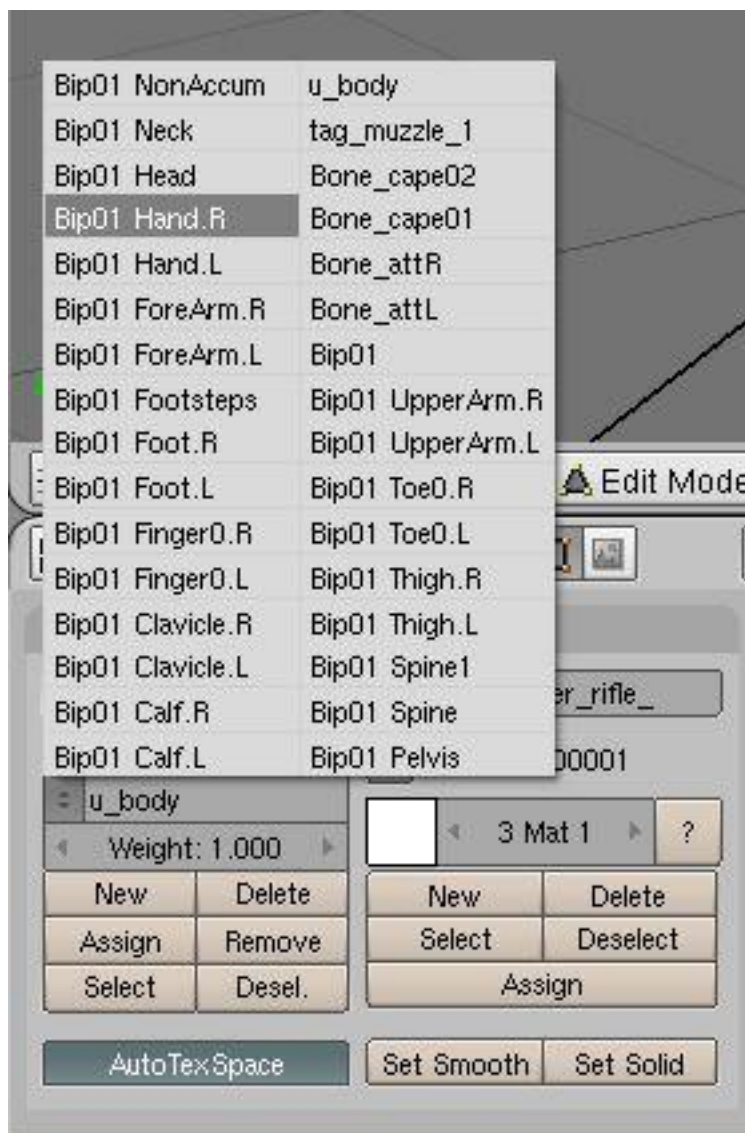




Let's direct them to the group Hand.R. For this in the section Vertex Groups (pic6\_3)



you need to select the correct group (pic6\_4)



and press Assign button(pic6\_5).

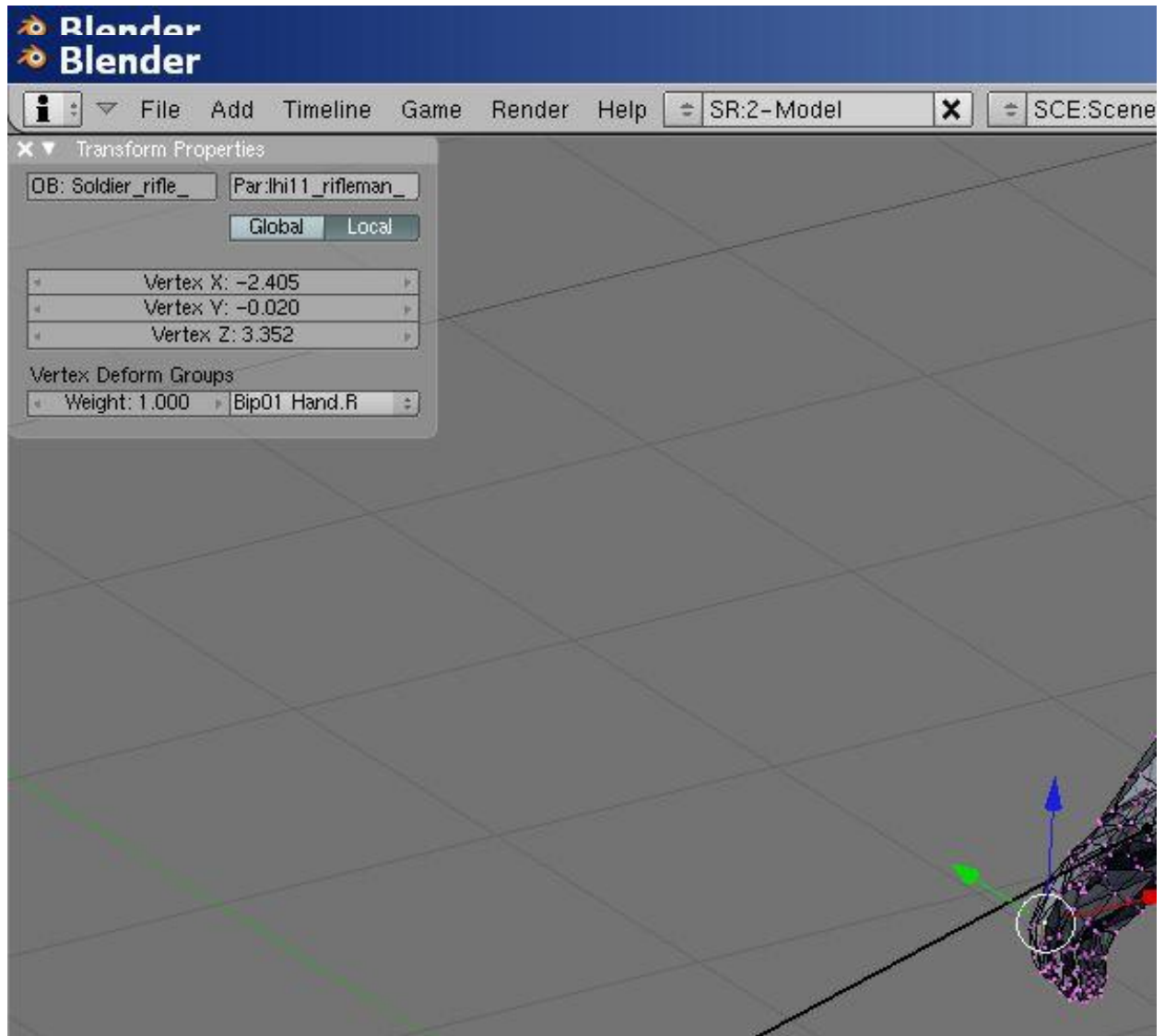


This way we assigned selected vertices to a given group.

Which vertices to which bone to constrain to can be checked in the original model. Import it somewhere nearby and watch it,

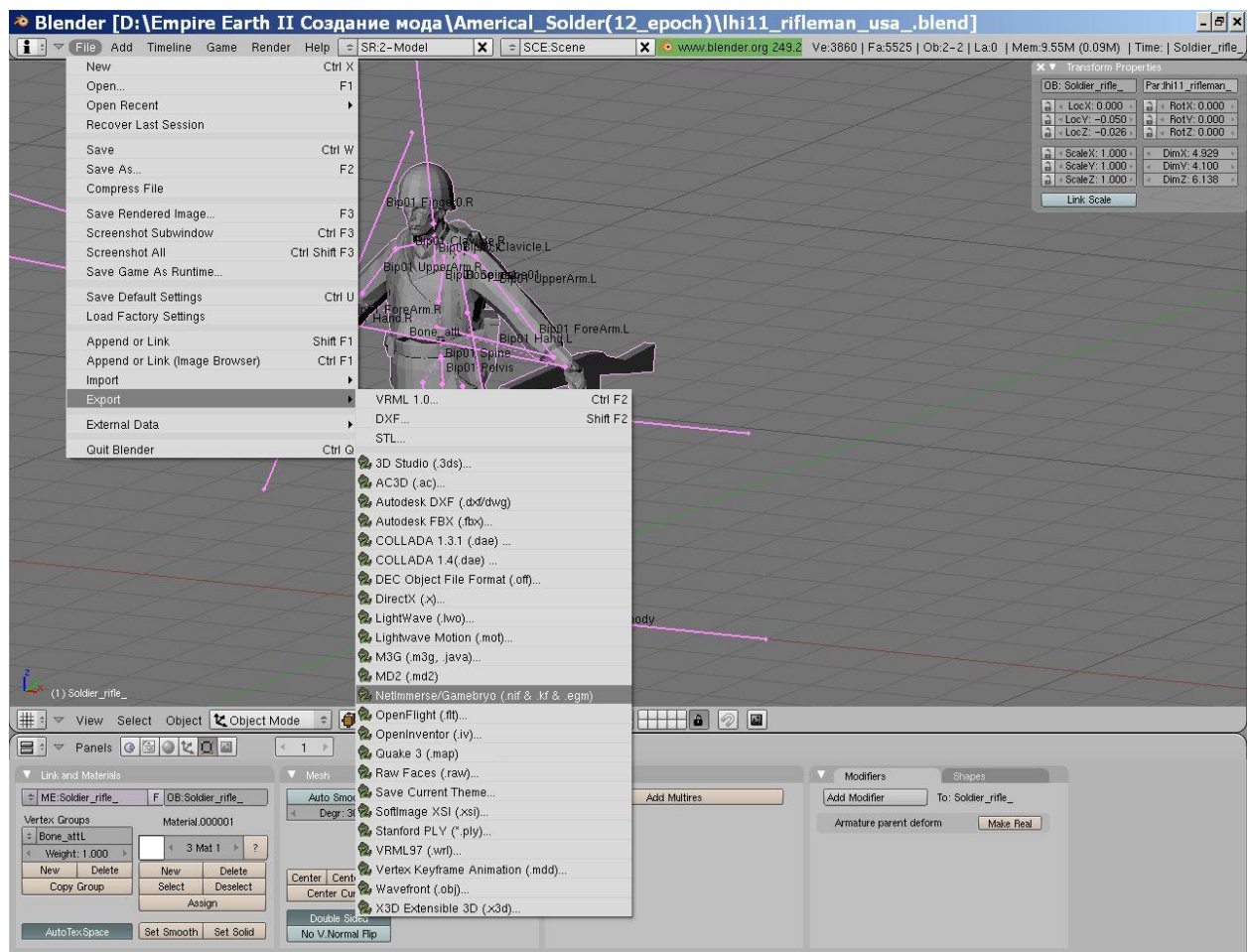
and when you're done, delete it.

If you turn on properties (key N) select one of the vertices just attached and then you will see which group it belongs to (pic6\_6) and whether he belongs to any group at all. Looking back, I will say that there should not be such vertices, everyone should be assigned to something.



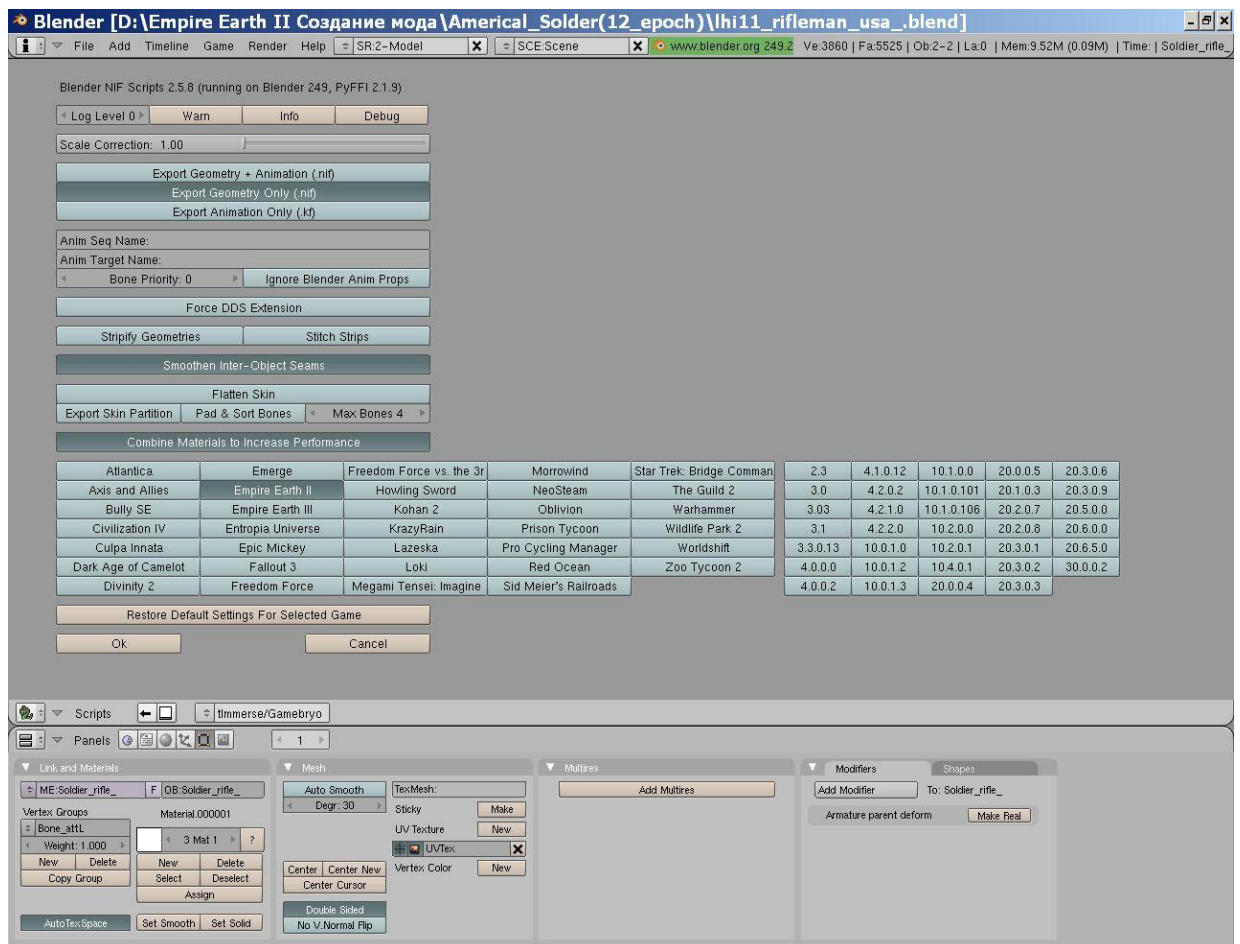
Therefore, all vertices must be connected.

After joining all the vertices, exit the edit mode (key Tab), select all (press A until all objects are selected) and export the model in .nifcache format (pic7\_1).



With the settings as in the picture (pic7\_2).



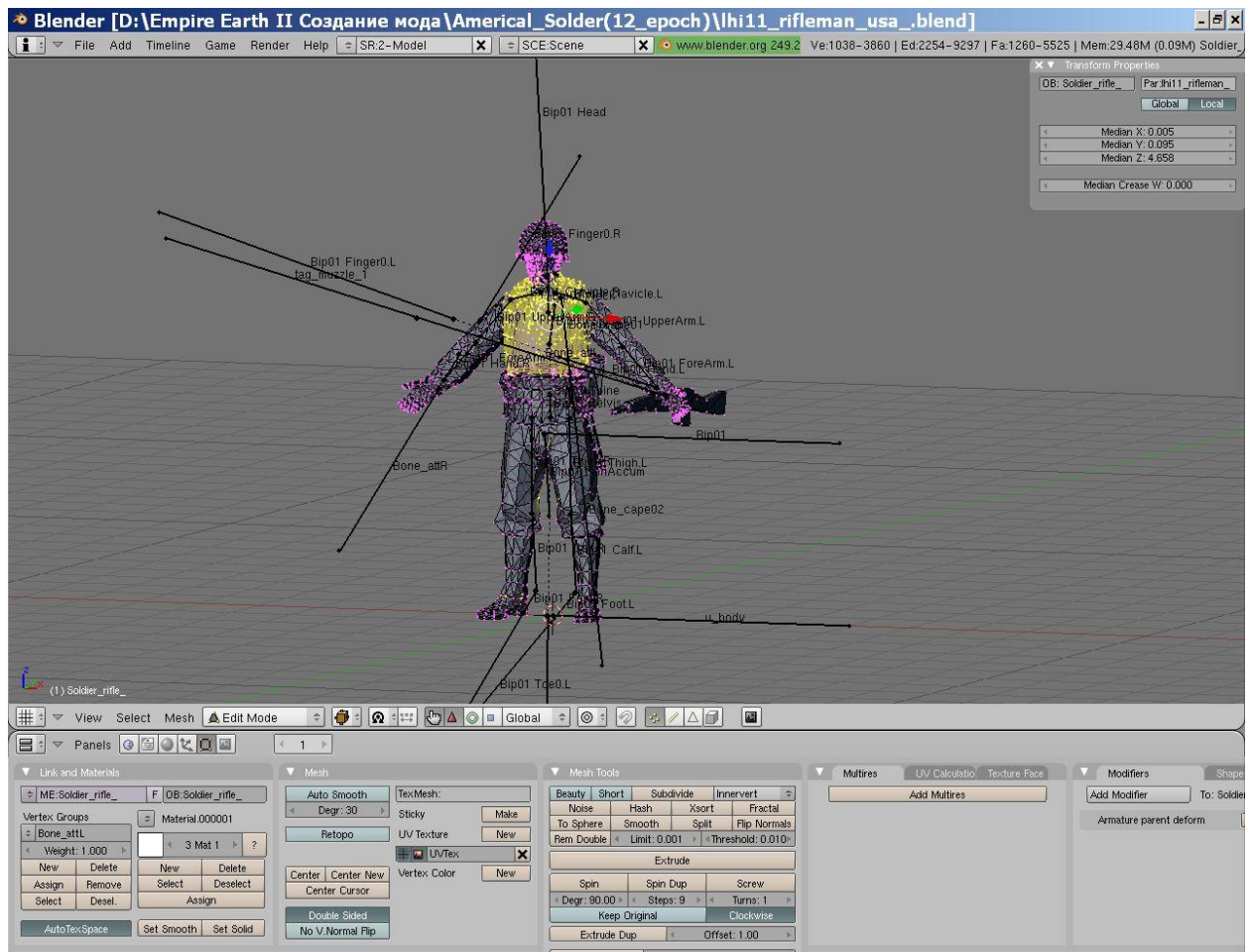


If an error occurs during export as shown in the image below (pic7\_3),

#### EXPORT ERROR

Cannot export mesh with unweighted vertices. The unweighted vertices have been selected in the mesh so they can easily be identified.

this means that some vertices were not attached to the bone. If you click on the message, the model will be displayed in edit mode and problem vertices will be highlighted (pic7\_4).

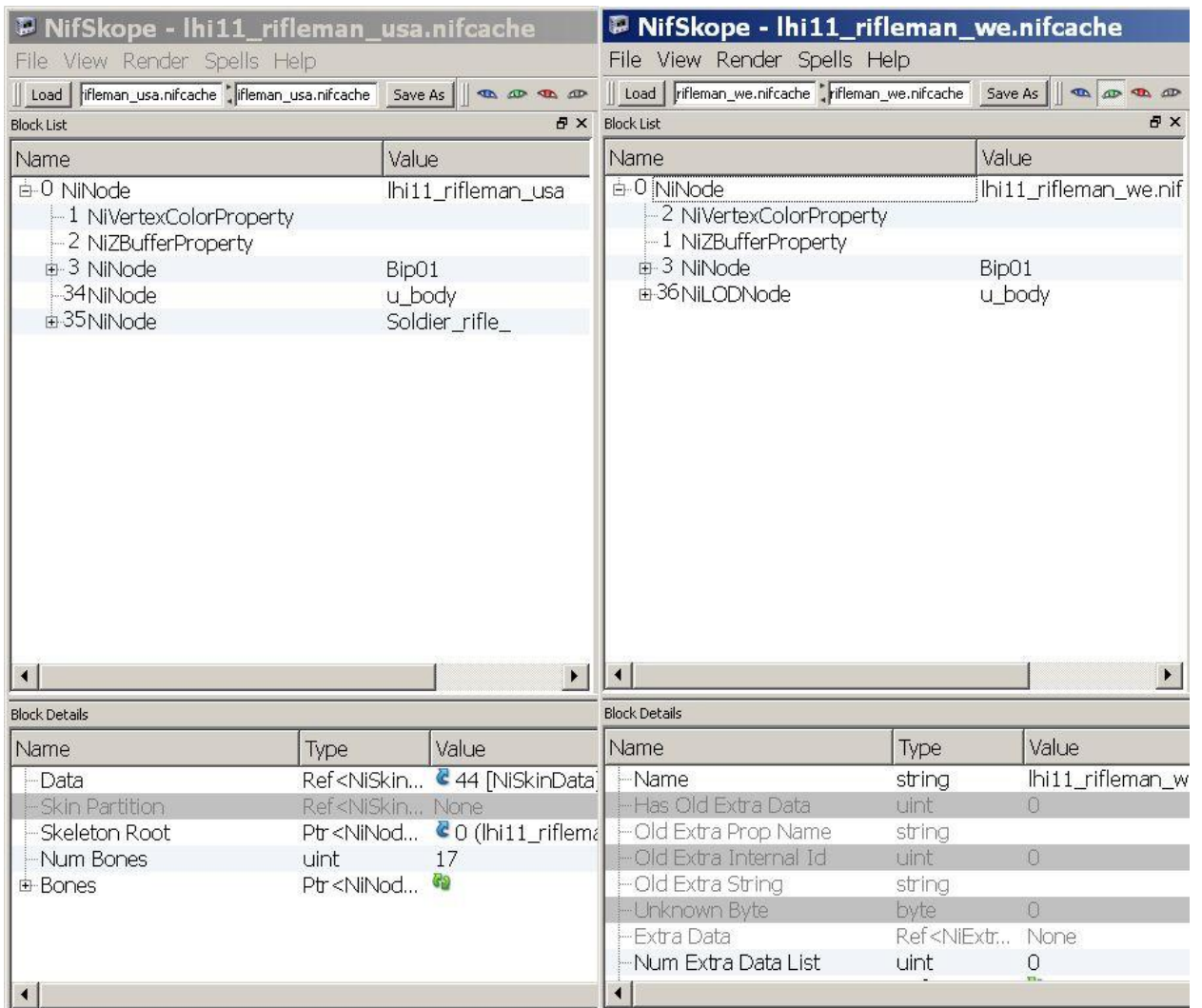


If no errors occurred, wait while the model is exported. This may take from a few seconds to several minutes. The red bar will gradually turn green (pic7\_5).



Open the final model and the original in NifSkoPe and compare (pic8\_1). The differences are immediately visible.





We will delete the empty node named u\_body i węzeł Soldier\_rifle\_ rename to u\_body.

Let's go through all the nodes.

Be sure to compare all nodes to the original model.

Let's check the operation of the model. To do this, rename it to lhi11\_rifleman\_we.nifcache and put it in the game folder.

Texture is not displayed in-game. Most likely, this is due to the fact that we immediately imported the model into Blender from the .obj format. Before that, you can try to import it to NifSkope and save it in .nifcache format, and in Blender we already import .nifcache.

As the process has not yet been fully tested, such issues may arise. Therefore, you can entrust the automation of vertex binding at the beginning, so as not to waste time. See what the model will look like in-game. Overall, will this work and whether textures display normally. If everything is fine, you can already manually tie. The only thing after snapping in automatic mode is that some vertices may be out of place. (pic9\_1).



*An important addition*

*Improving the method*

And now an important addition:

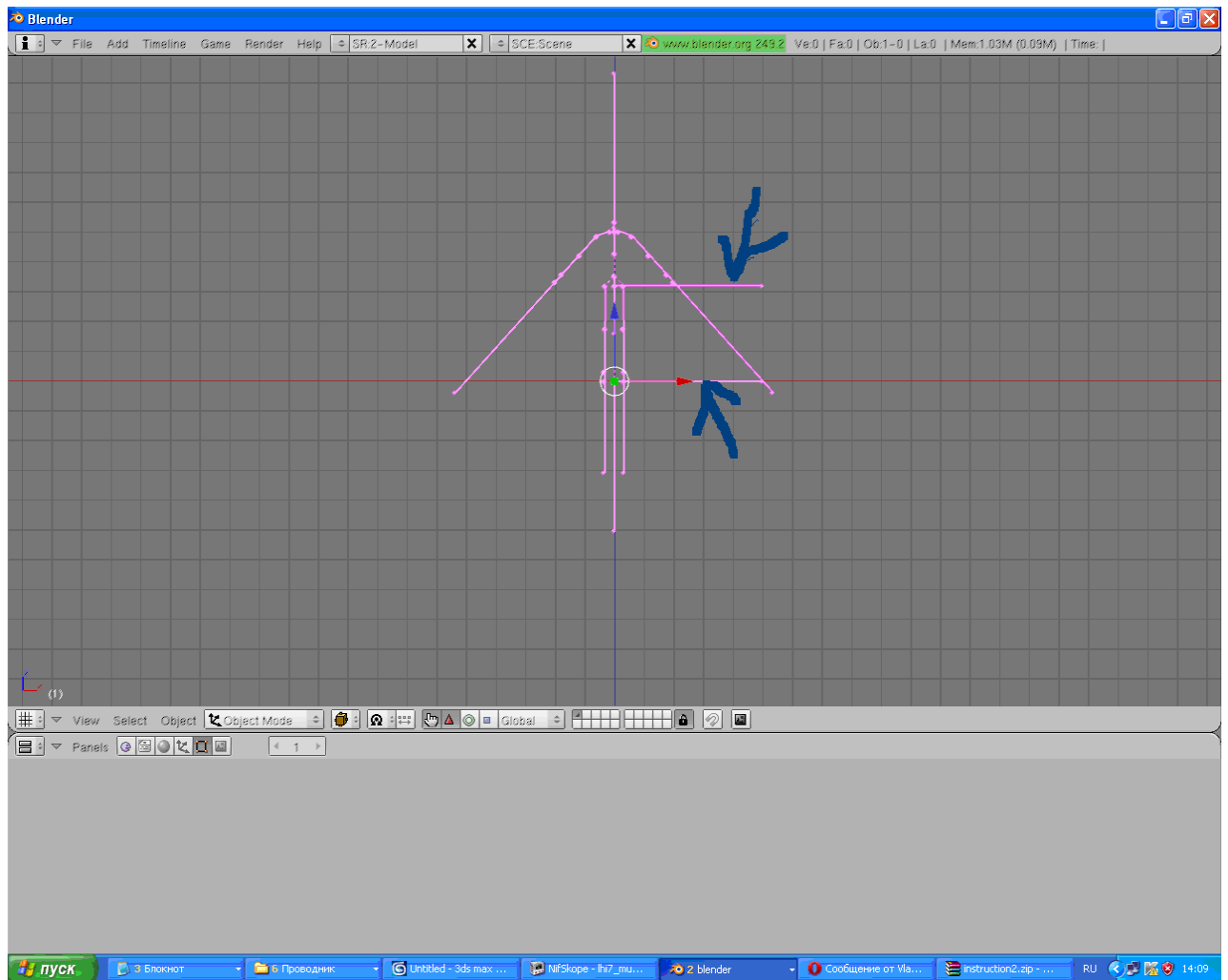
This method has been greatly improved, it has become faster (20 minutes to adjust one model) and less labor-intensive - constraining vertices is assigned automation.

The essence of the improvement is that the constraint of the vertices is completely committed to automation. This has two undeniable advantages:

- 1 Automation does the job in 5 seconds. And it takes a human being much longer, and then it takes a long time to correct it.
- 2 Automation does the job better than humans.

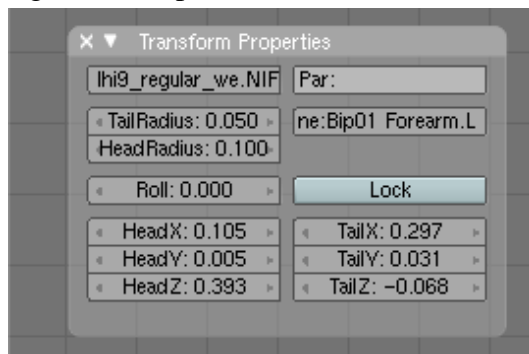
If you automatically constrain the model to a raw skeleton, many vertices do not connect correctly, and the arms and legs may not be in place. The problem is that in the original skeleton, apart from the actual bones, there are many other fittings responsible for weapons, moving parts and auxiliary fittings. They all intersect or are close to the major bones of the skeleton, and automation often confuses these skeletons with bones, leading to misalignment and ultimately disfiguring the model.

For example, in this skeleton, only two armatures interfere with normal model constraint:

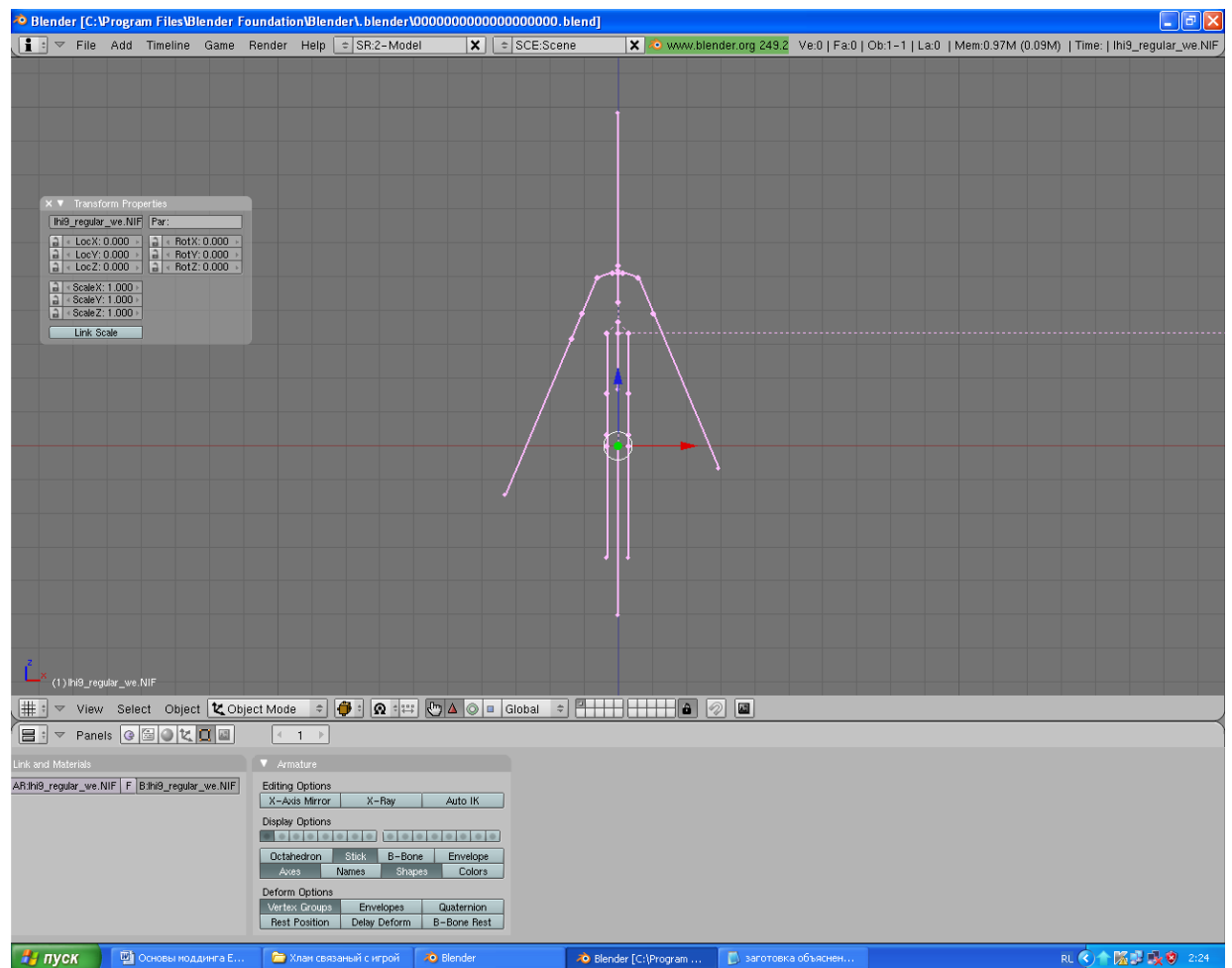


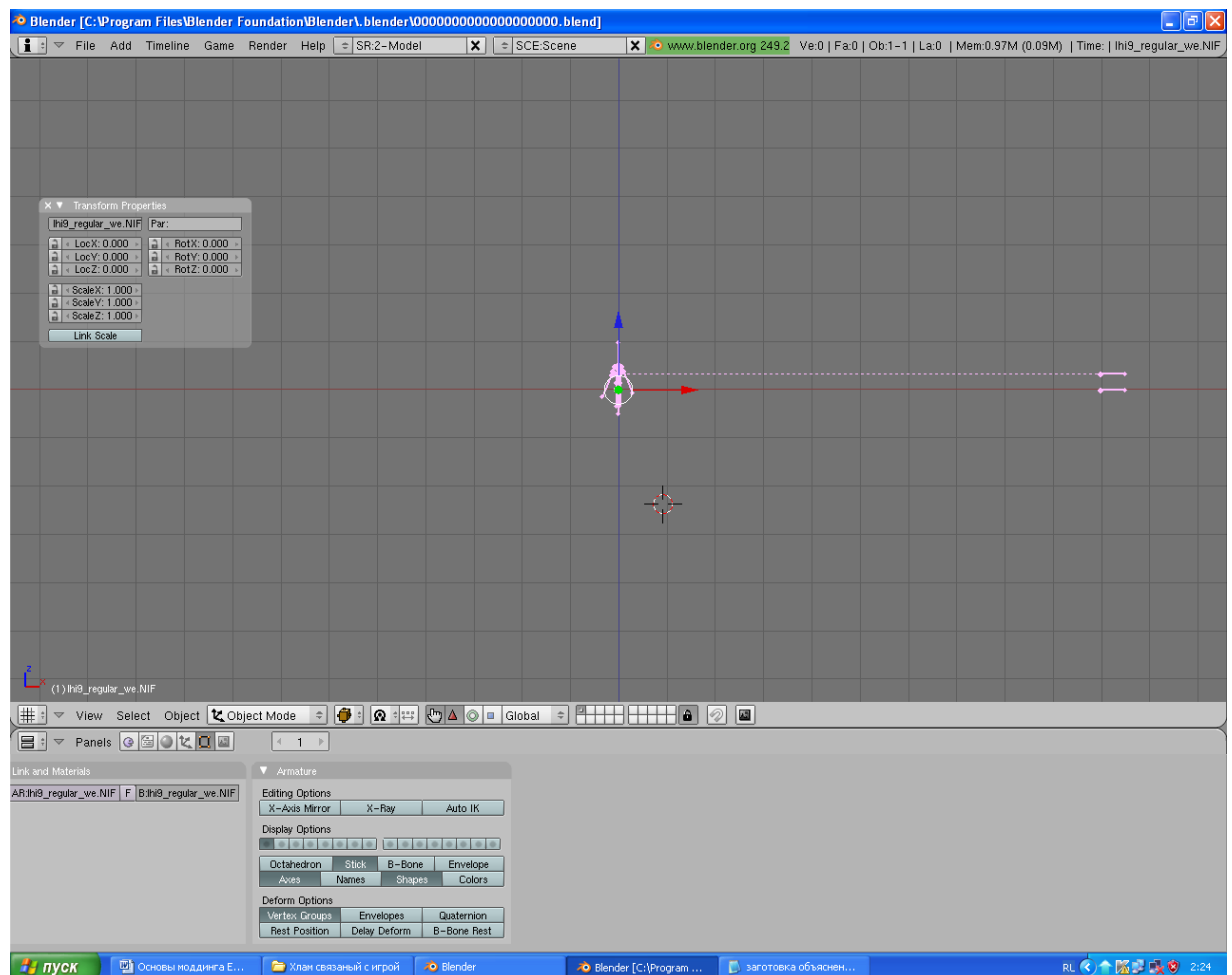
These parts can be easily pushed back during snapping, when you snap the model to the armature, the armatures are returned to their place.

To move an armature, enter the edit mode (Tab), select the required armature with the right button, press N, the familiar menu will appear:



Increase the HeadX and TailX parameters to 10, in this case, the reinforcement without distorting its size will be moved by a distance of 10 from the model and will no longer interfere with the automatic tightening. Then, after returning the skeleton to its place, simply reduce these parameters by 10.





This way, by moving the harmful reinforcement away from the frame, you will get a perfect connection between the model and the frame. Different skeletons can have a different number of additional skeletons, usually from 2 to 5. A horse's skeleton can have 10 such skeletons.

Some possible problems and how to solve them:

1. If an error occurs during autobinding:

This means that there are parts of the model that extend far from the skeleton. The skeleton and the model must match. It is very important that they are the same size. They were in space within themselves (bones in flesh). Everything should be like in life. The bones should not be visible. Sometimes the cause of the error can be such parts of the model as wide-brimmed hats, feathers on the hat, threads on clothes, shoulder pads, parts of the armor, a sword scabbard, etc. (if they stand out strongly from the body). which parts should be deleted or exported separately already in Nifskope format.

2. If the model is not saved, individual vertices will not be included. Automation sometimes forgets to snap individual vertices or even parts of the model. They are mostly the same parts of the model that stand out from the body: wide-brimmed hats, feathers on the hat, threads on clothes, shoulder straps, parts of the armor, a sword scabbard, etc. They have to be tied by hand. If the reason is one part of the model - for example just the cap - then immediately after looking at the bug (you have all unattached vertices marked at this point), you can freely tie it to your head - you don't need to tie all vertices one by one.

Useful tips for Blender beginners:

- 1.Models and dice in a blender are selected not with the left, but with the right mouse button.
- 2.To rotate the model in the blender along one axis, press R, then one of the three X Y Z keys.
- 3.To rotate the model as you want, just click R.
- 4.To move a model in space, right-click the model and, without releasing the right mouse button, move it in space.
- 5.To reduce / enlarge the model - select it, then left-click on an empty space in the blender window and, without releasing the left mouse button, move it down.
- 6.Delete Cube, Camera and Light - Right-click these items then click delete.
- 7.Always save when working in a blender. After removing all harmful parts from the skeleton - remember to save - this will be the template for you if you want to customize more models.
- 8.To see which bone the vertex is attached to, use the N button to call up the info window. Also in this window you can change all the numbers, for example the weight of the vertex connected with each bone.

Sometimes in the original models, the arms are stretched along the horizon and the legs are spread wide. To fit such models to the skeleton ee2 - select an arm or a leg and rotate (if necessary and move it) under the skeleton. For example, models of people from civ4 with arms outstretched along the horizon and legs wide apart. Models from Rome Total War (used in the EE4 mod) only with outstretched arms along the horizon. To select a part of the body for subsequent rotation, it is much more convenient to select it not with vertices, but with triangles (polygons).

I have not noticed any problems with the node names in Nifskope after exporting from the blender. I think there is no need to change the name.

Sometimes it happens that everything is fine with the model - it walks, it shoots in the game, but it has no texture and it cannot be installed. To make the final model immediately textured, import the mesh into the blender not from 3ds or obj files, but only from nif format. Therefore, run a working nif file into which you will first import your 3ds or obj models, and then from that file into the blender. Also, don't forget to specify the mesh textures in the working nif file in the standard Nifskope fashion.

P.S.

All animals are made in the same way as humans, you just need to choose the right skeleton from the original game skeletons.



## 8 How to add player color bars.

Author P.M.A.



### Adding player color bars

By colored bars we mean those areas of a unit or building that are painted in the color of the player they belong to.

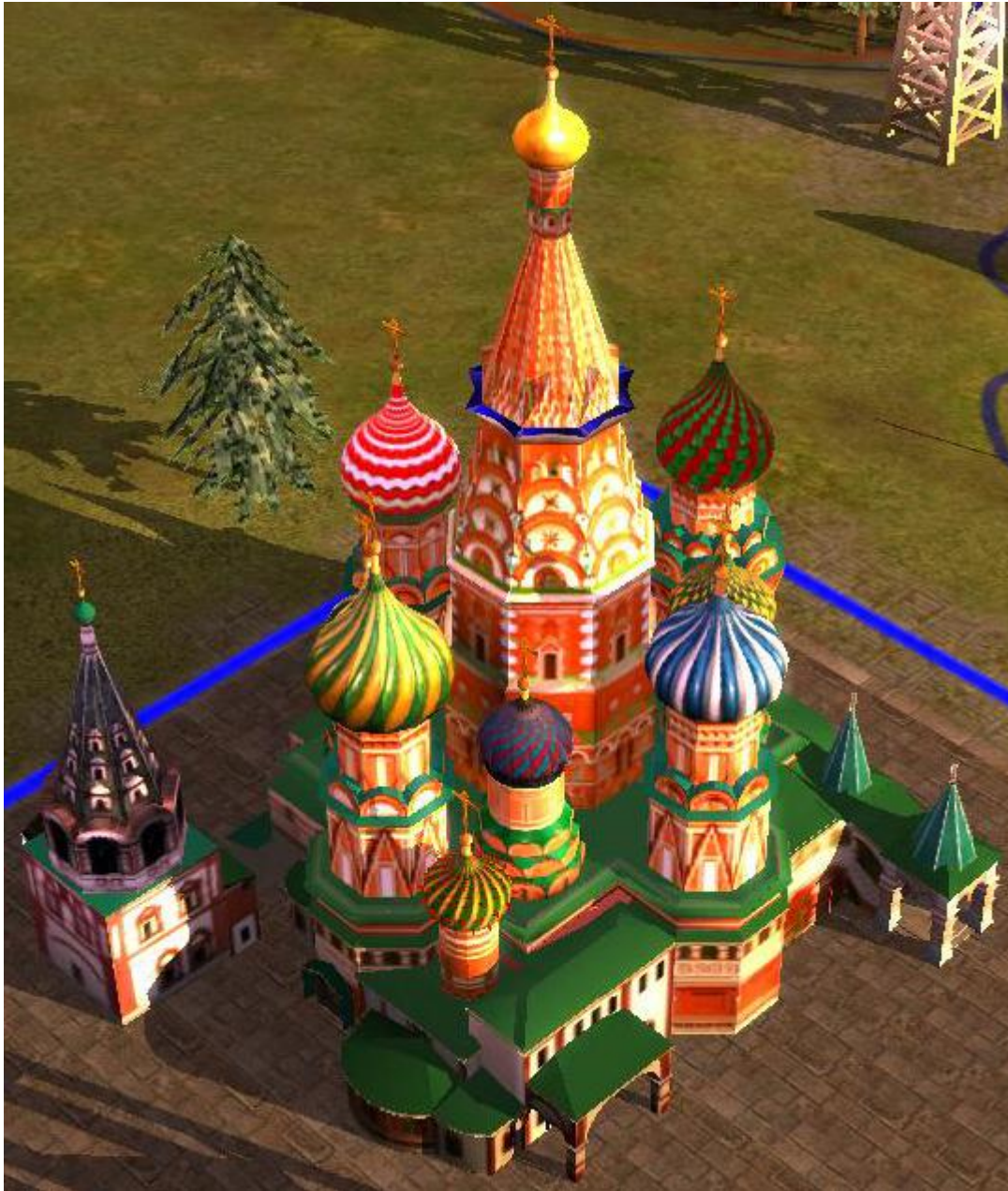


In the original game, all units and buildings have these bars. This allows players to better navigate.

The developers created the following set of colors:



So let's start. Let's add colorful stripes to the new building, namely: the Pokrovsky Cathedral. We did it as a Large Building (wonder) for Russian civilization instead of the Brandenburg Gate.

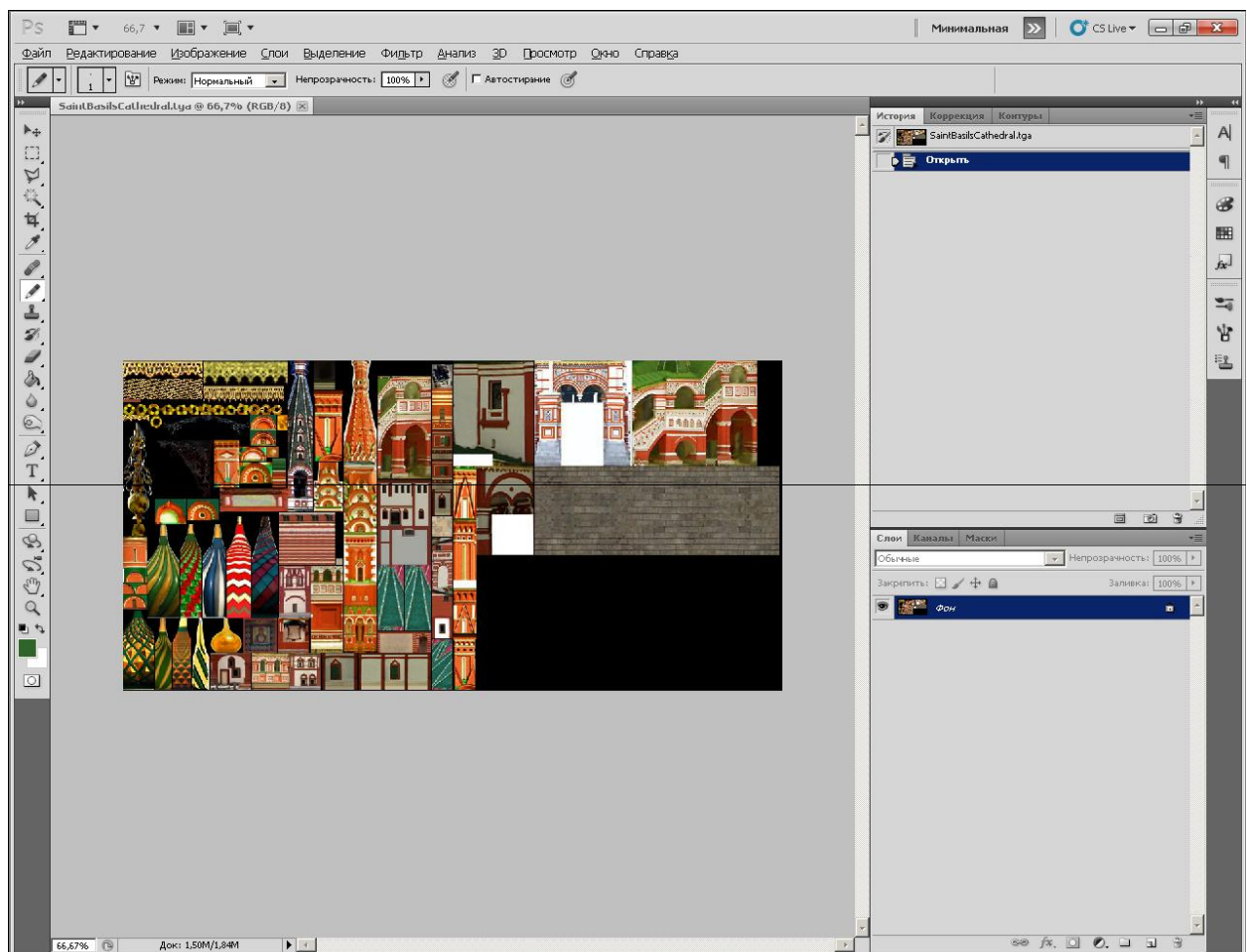


The picture shows that the periphery of the central tower is blue. This is our color bar. Next, step by step, its creation will be described.

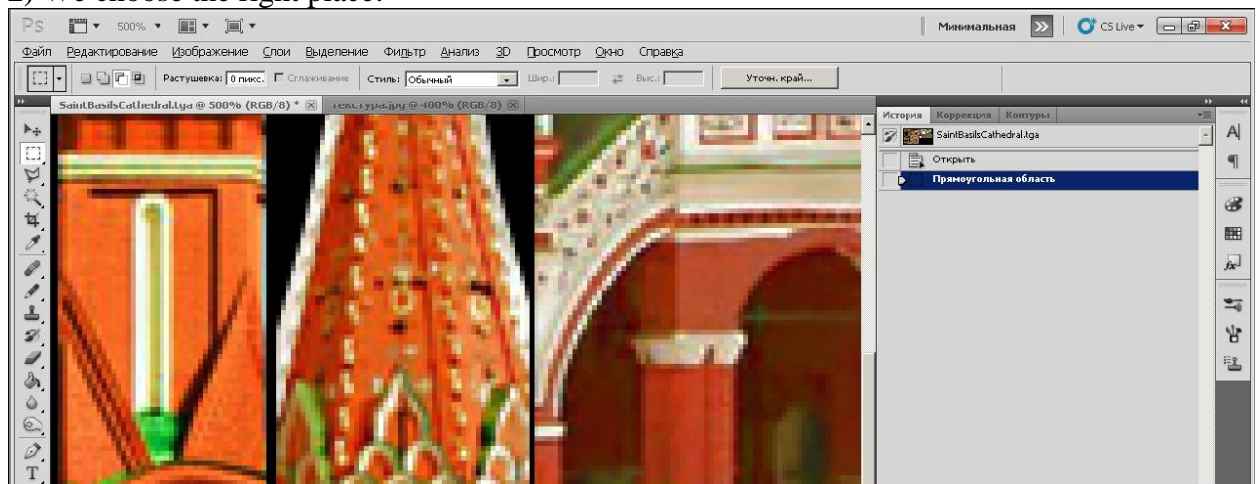
We need a texture that holds that part of the building, in our case the cathedral only has one texture - **SaintBasilsCathedral.tga**

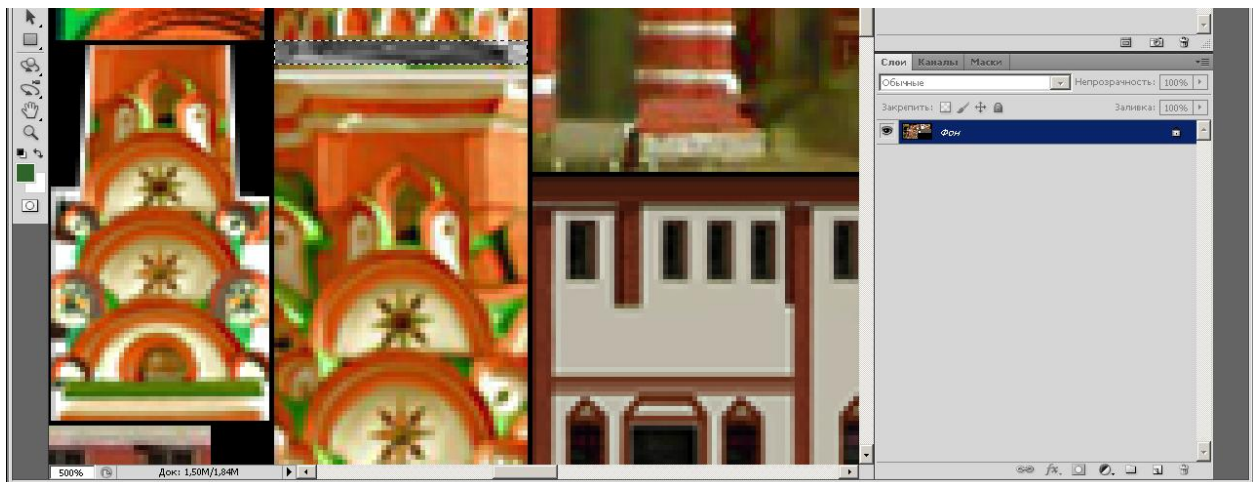
1) We open it in the program **Adobe Photoshop**:



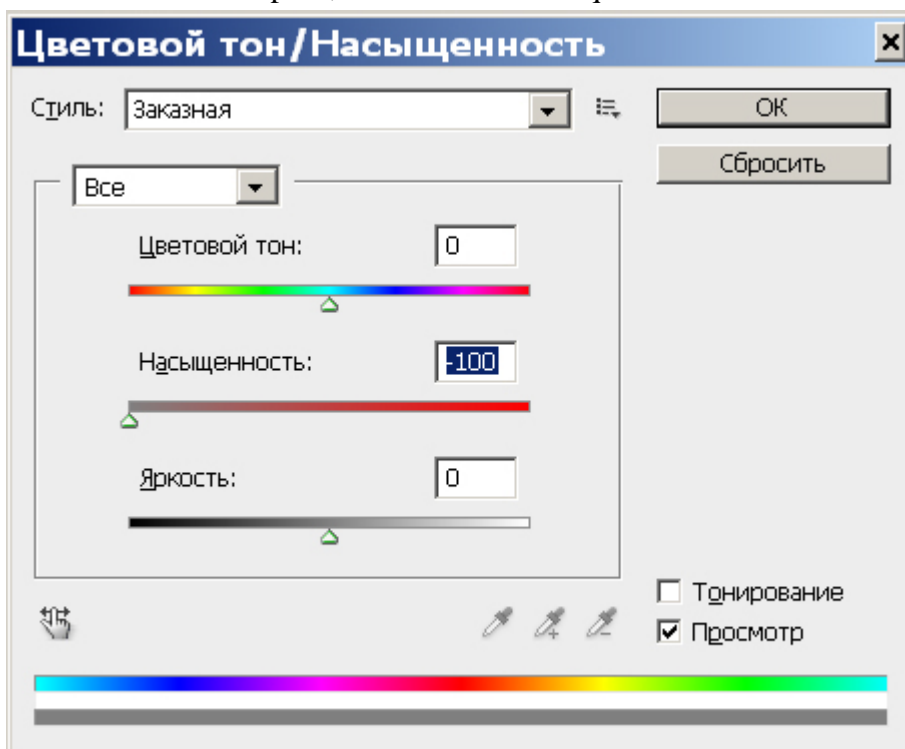


2) We choose the right place:



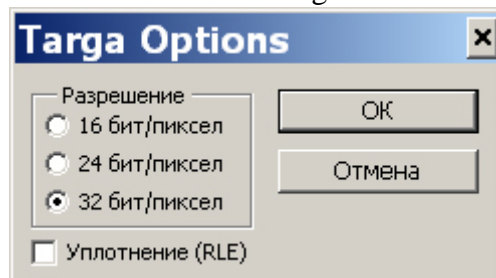


3) We press **Ctrl + U** or **Menu ▾ Picture ▾ Correction ▾ Color tone / saturation...**  
In the window that opens, set the saturation equal -100:



and press OK.

4) All. We save it in the **.tga** format with these settings:

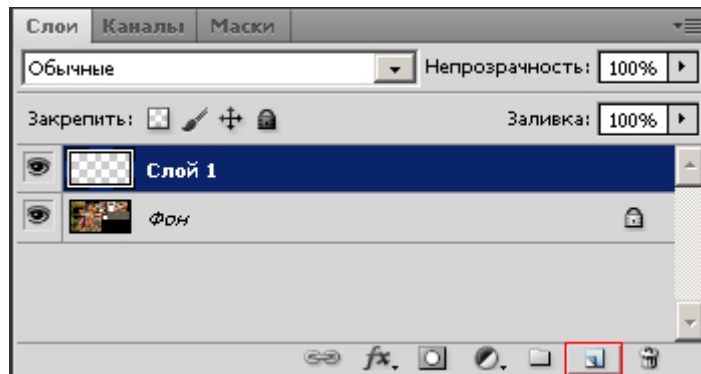


Then we need to create a texture with the same name, just add it to the end before unfolding - «**\_pc**» (Latin, without quotation marks). This means that we will be successful - **SaintBasilsCathedral\_pc.tga**. This texture should be the same size (width and height) as the original (**SaintBasilsCathedral.tga**), so to make it easier we just rewrite the original and save it under a new name.

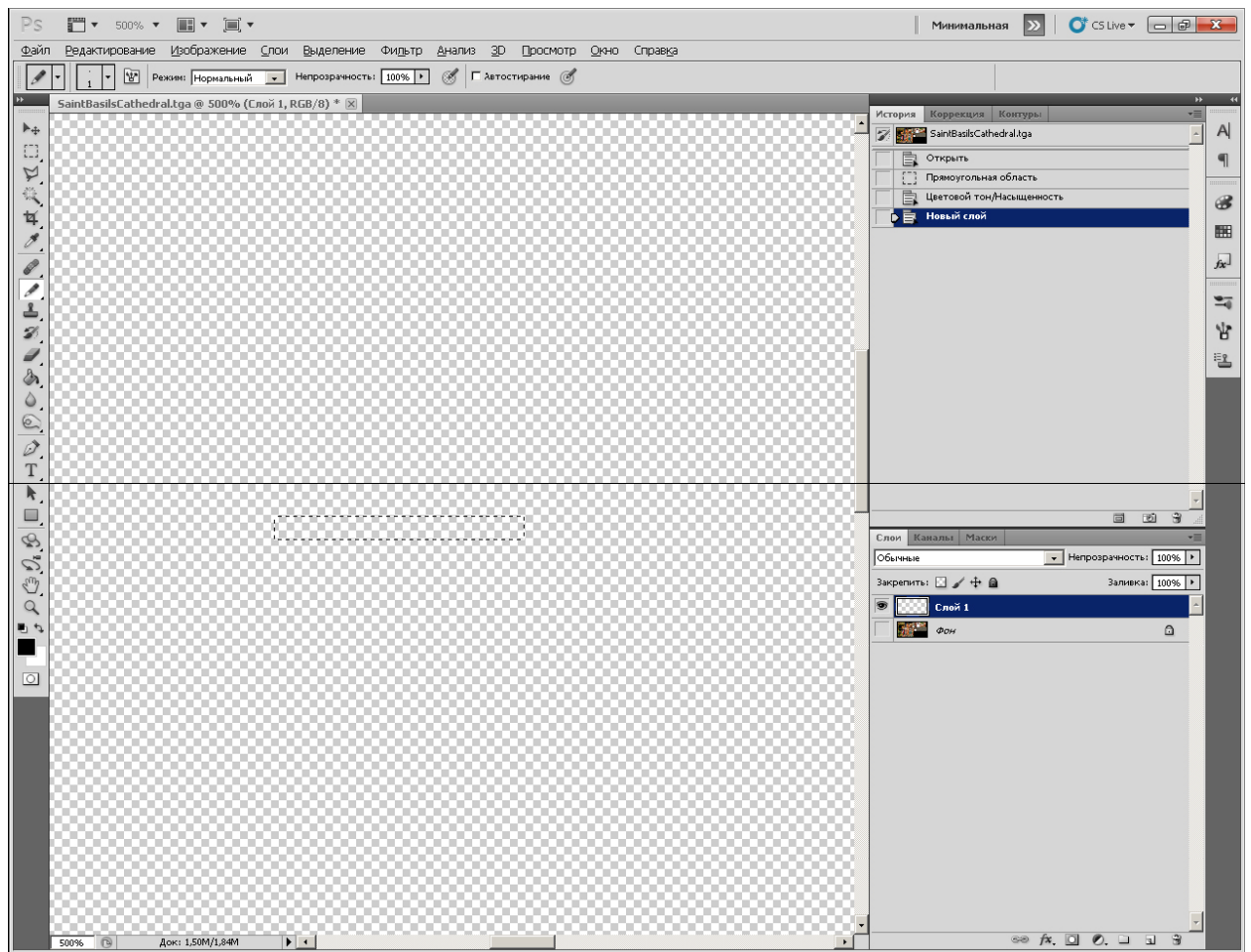
Before going to step 5, make sure you have Photoshop turned on, open the texture **SaintBasilsCathedral.tga** and select the area we removed

saturation.

5) Create a new layer. To do this, press the key combination **Shift + Ctrl + N**, and in the window that appears, click OK. Or press the button marked in red in the picture:



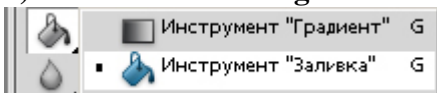
6) Hide the background layer. To do this, remove the eye icon next to it or select the layer and click **Menu → Layers → Hide layers**. It should look something like this:



7) Choose a primary color **white**:



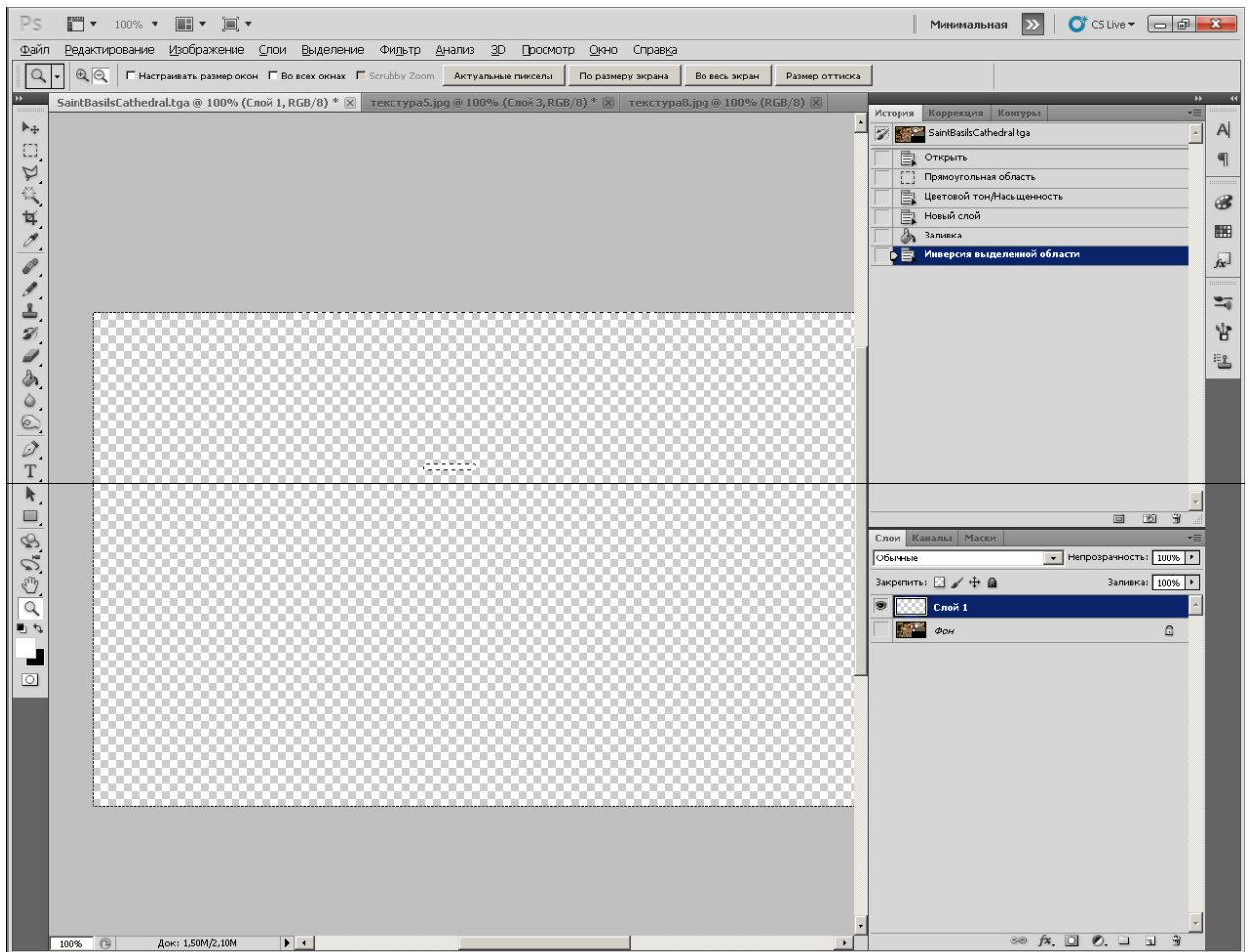
8) Select a tool «**Filling**»:



9) Fill in the selected area:



10) Invert Selection. To do this, press the key combination **Shift + Ctrl + I**, or **Menu → Selection → Invert**:

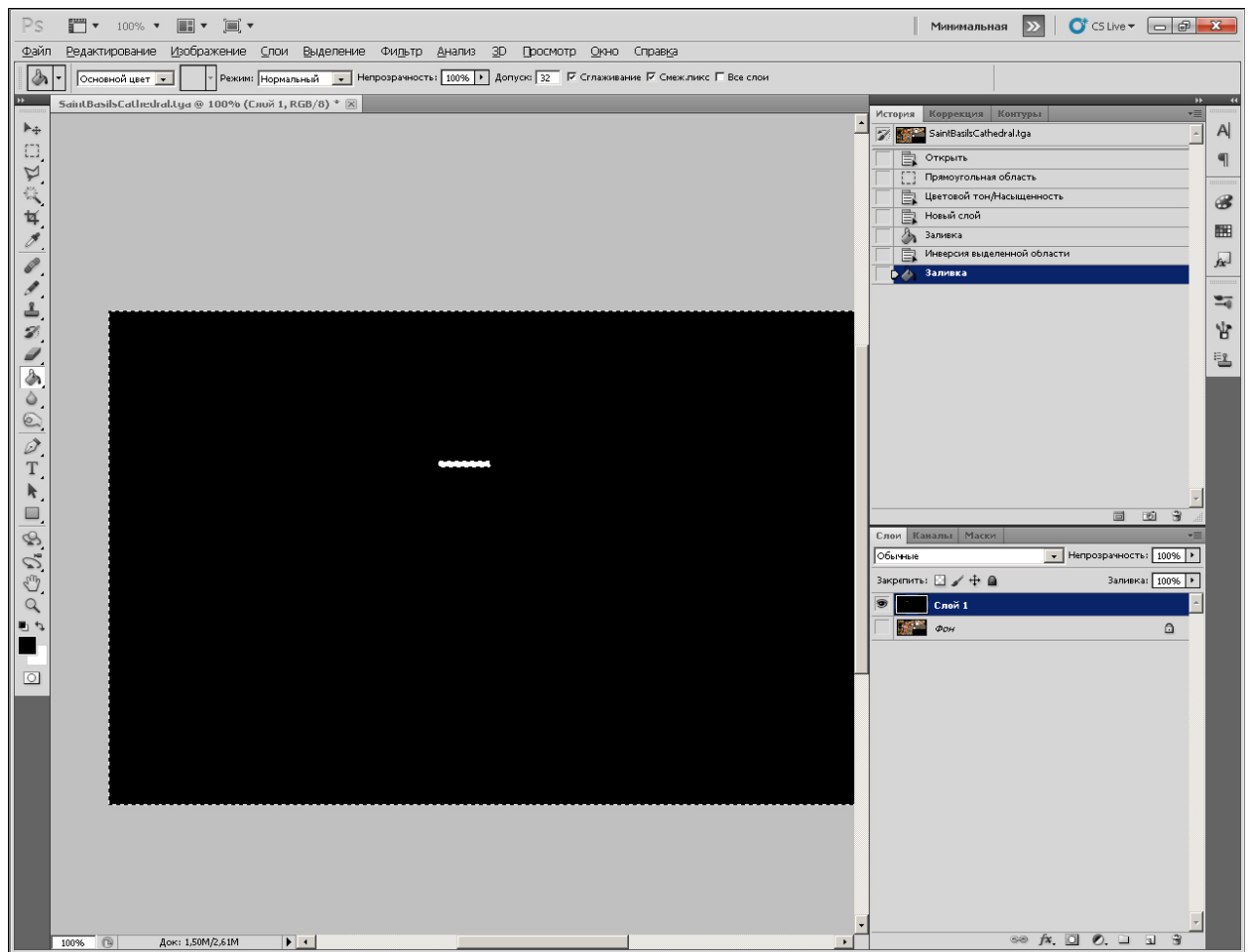


11) Choose a primary color **black**:

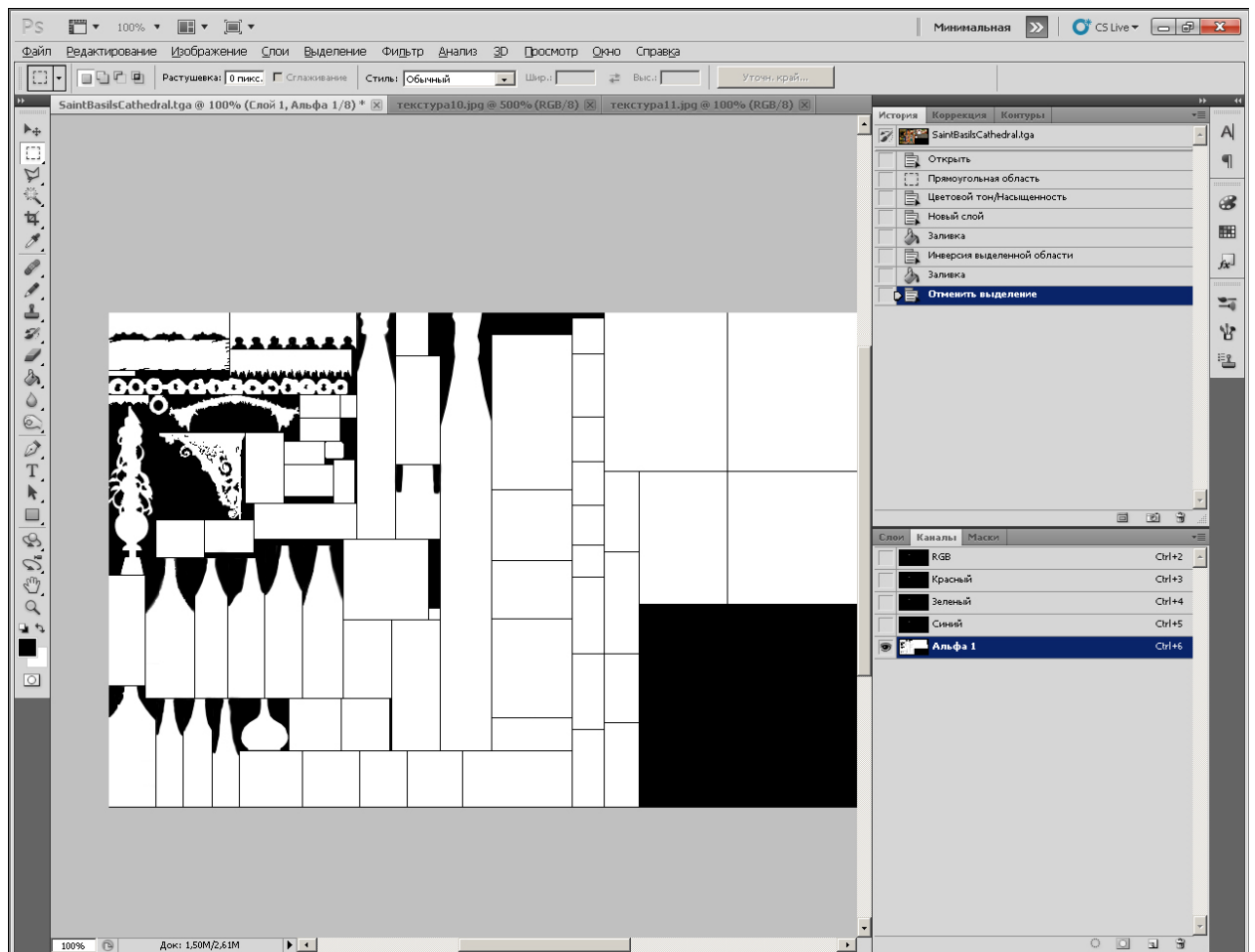


12) Use it to fill in the marked field:



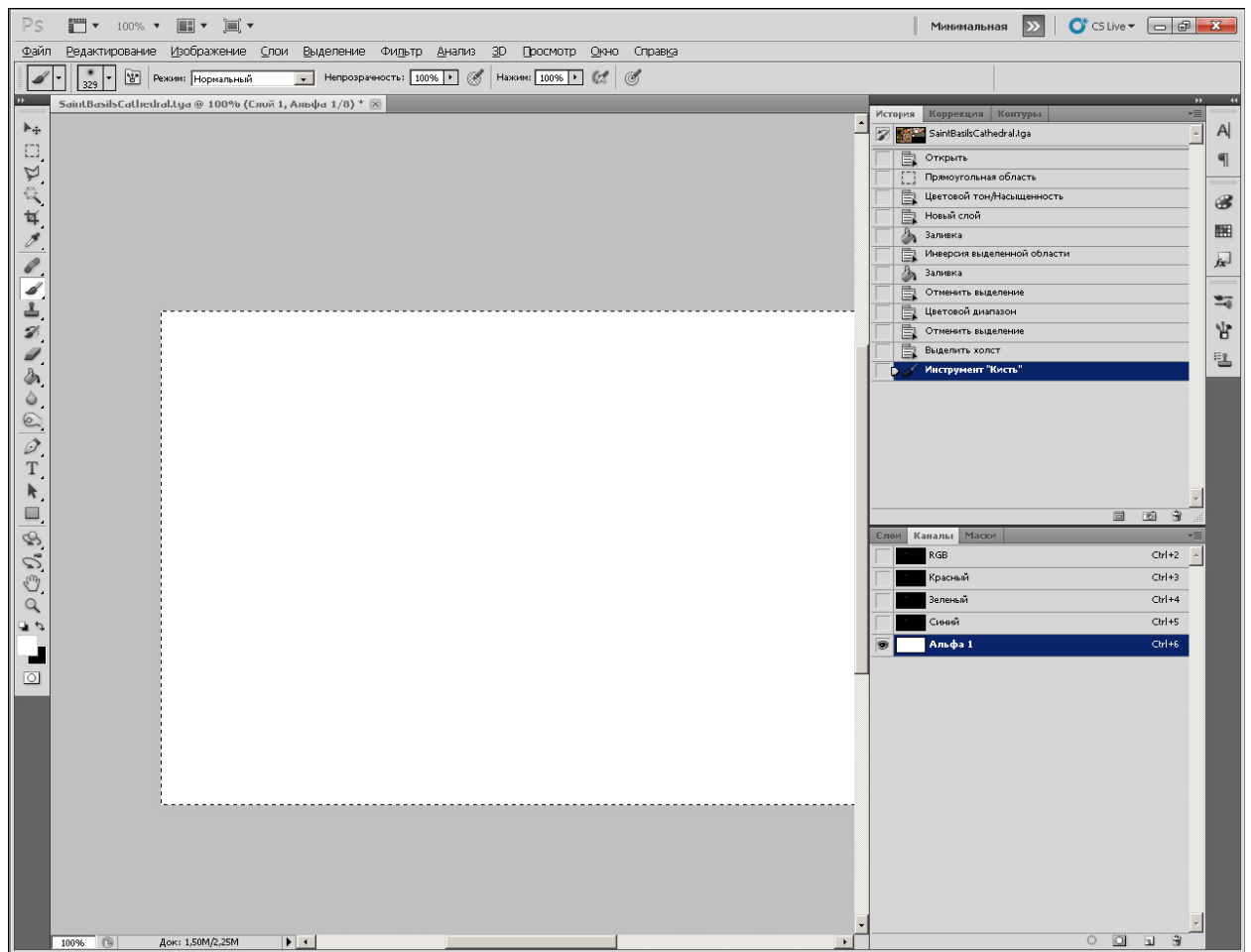


**13)** Deselect the check mark (**Ctrl + D**, or **Menu → Selection → Cancel choice**) and go to the tab «**Channels**», if not, turn it on **Menu → Window → Channels**. Choose channel Alpha (if not, create it by clicking on a button similar to the button with which we created the new layer):



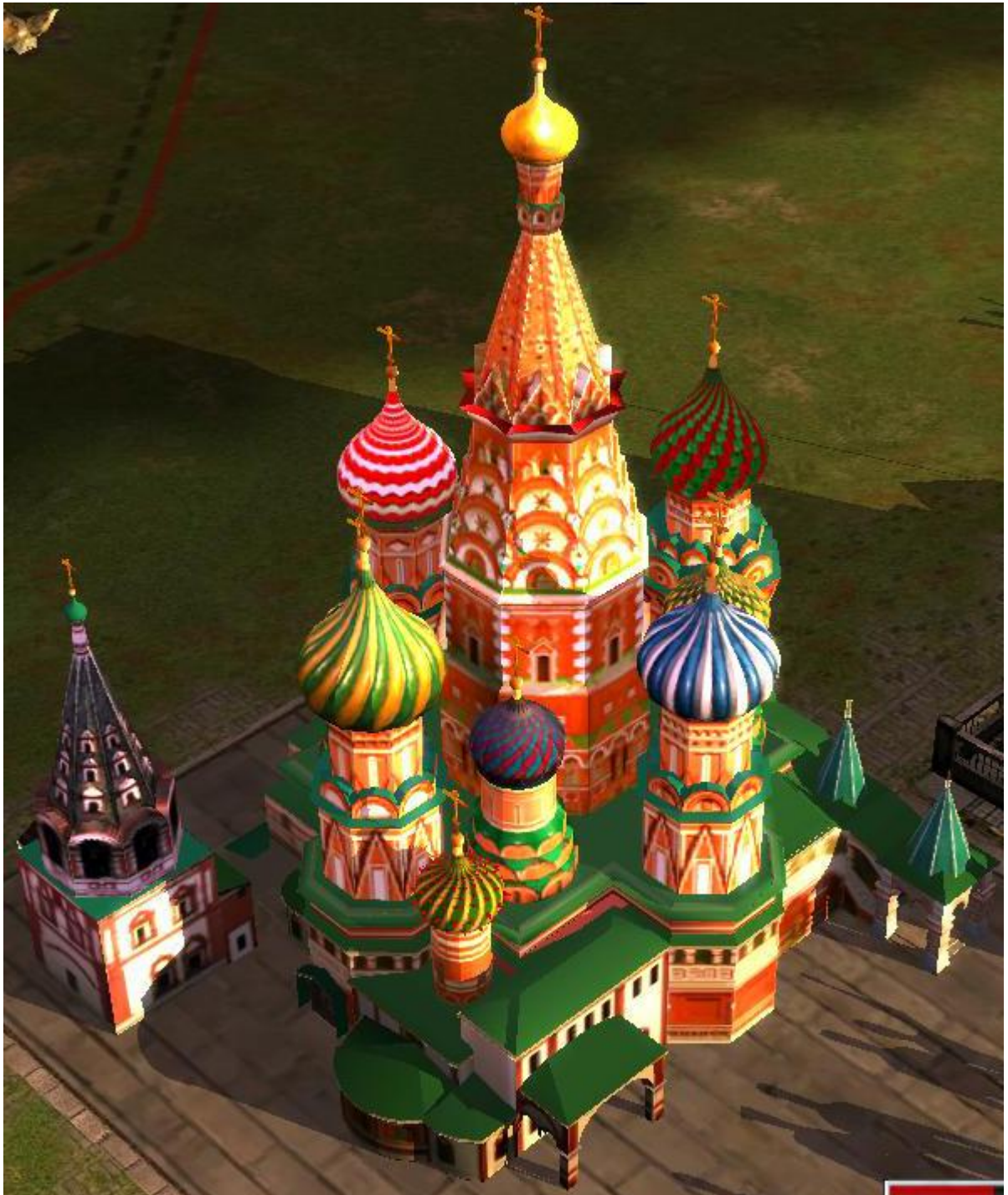
The alpha channel controls the transparency of the in-game texture. The darker it is, the more transparent the texture.

**14) Paint the entire Alpha area on white:**



15) Remove the layer we hide and save the texture in **.tga** format under the name **SaintBasilsCathedral\_pc.tga**.

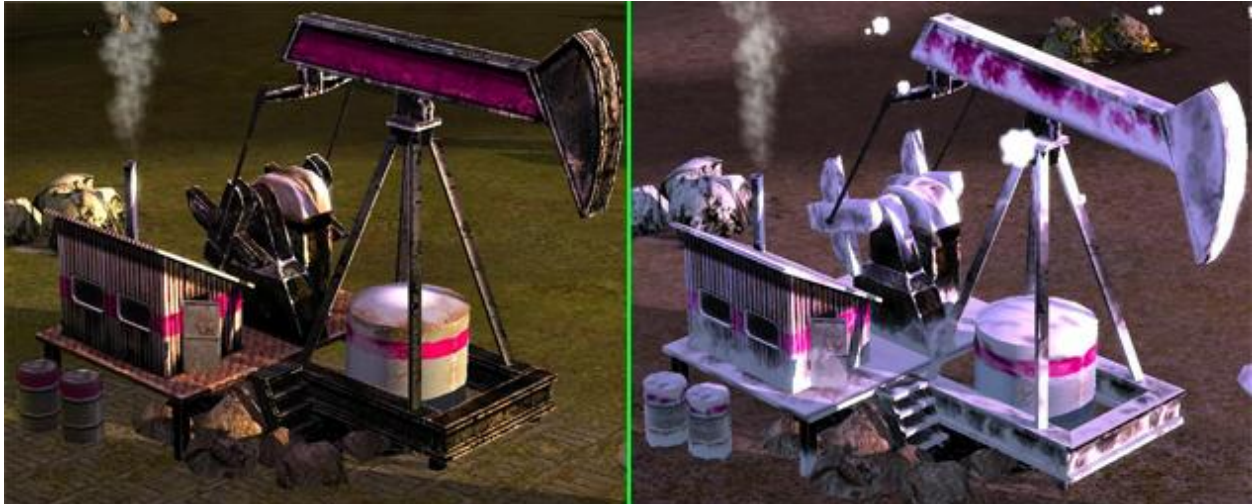
That's all. Now you just need to place these two textures - **SaintBasilsCathedral.tga** and **SaintBasilsCathedral\_pc.tga** - in the game folder and our cathedral will have its own distinctive strip:



## 9 How to add seasonal textures.

Author P.M.A.

In winter, all buildings are covered with snow in the game.



So that our cathedral is also covered with snow, let's draw a winter texture for it and write it under a name **SaintBasilsCathedral\_wi.tga**.

Then we need 3 files from the folder `\zips_ee2x\EE2X_db\Terrain`:

- **DbSeasonalTextureSets\_Arid.csv**
- **DbSeasonalTextureSets\_Temperate.csv**
- **DbSeasonalTextureSets\_Tropical.csv**

To files **DbSeasonalTextureSets\_Arid.csv** i **DbSeasonalTextureSets\_Tropical.csv** add the following line:

`SaintBasilsCathedral,SaintBasilsCathedral.tga`

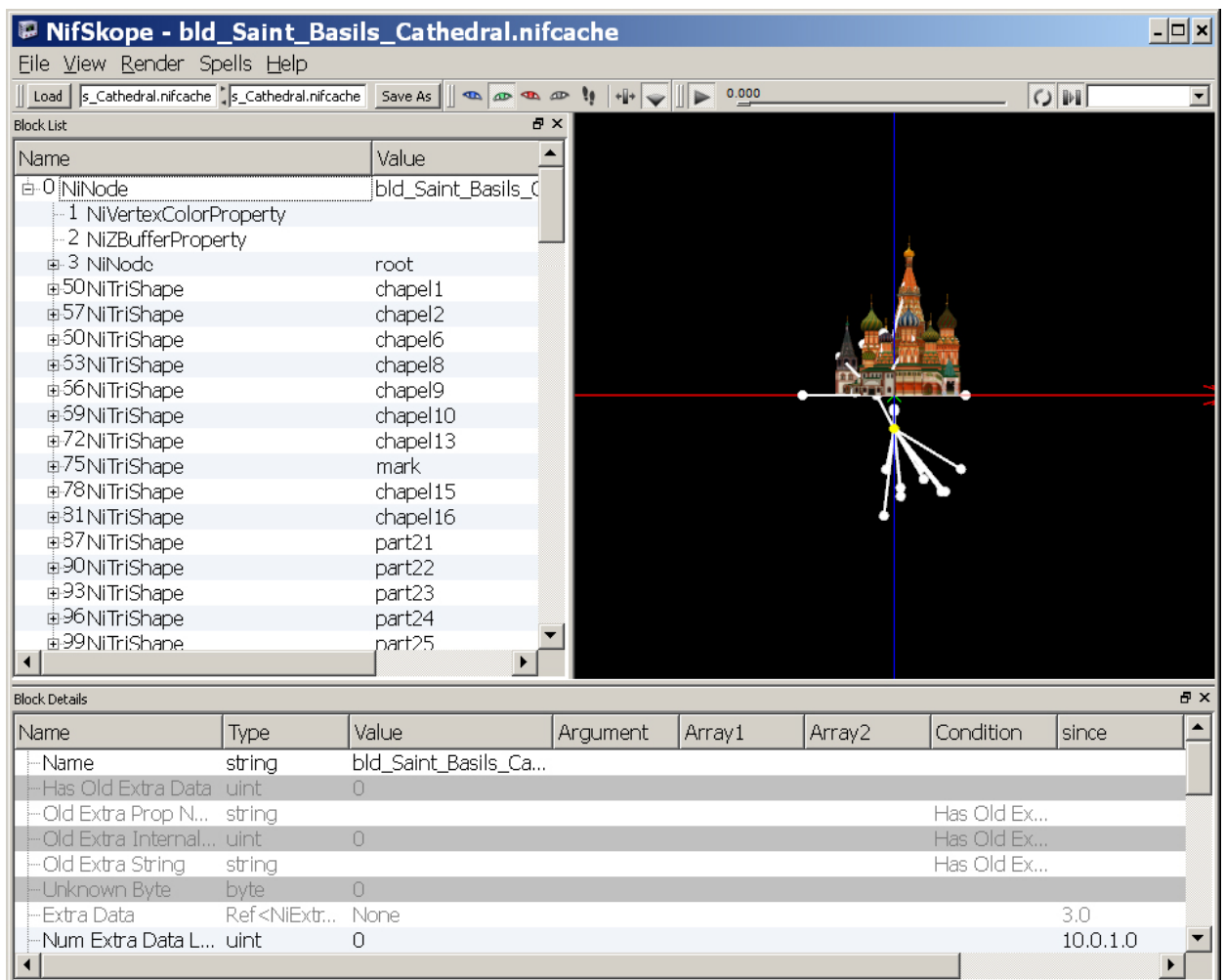
And to the file **DbSeasonalTextureSets\_Temperate.csv**:

`SaintBasilsCathedral,SaintBasilsCathedral_wi.tga SaintBasilsCathedral_wi.tga  
SaintBasilsCathedral_wi.tga SaintBasilsCathedral.tga SaintBasilsCathedral.tga  
SaintBasilsCathedral.tga SaintBasilsCathedral.tga SaintBasilsCathedral.tga  
SaintBasilsCathedral.tga SaintBasilsCathedral.tga SaintBasilsCathedral.tga  
SaintBasilsCathedral.tga`

We save the changes.

But it is not everything. Now you need to edit the model itself. We use the program for this **NifSko**pe. We open a cathedral in it:

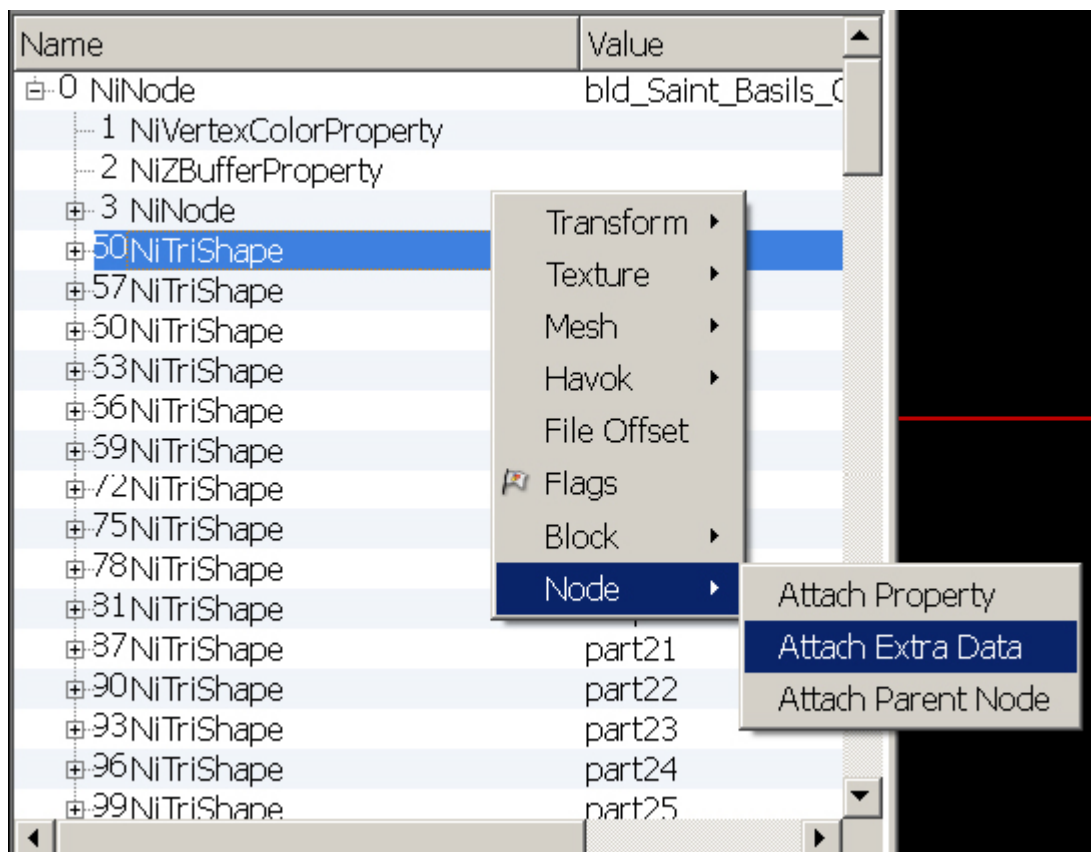




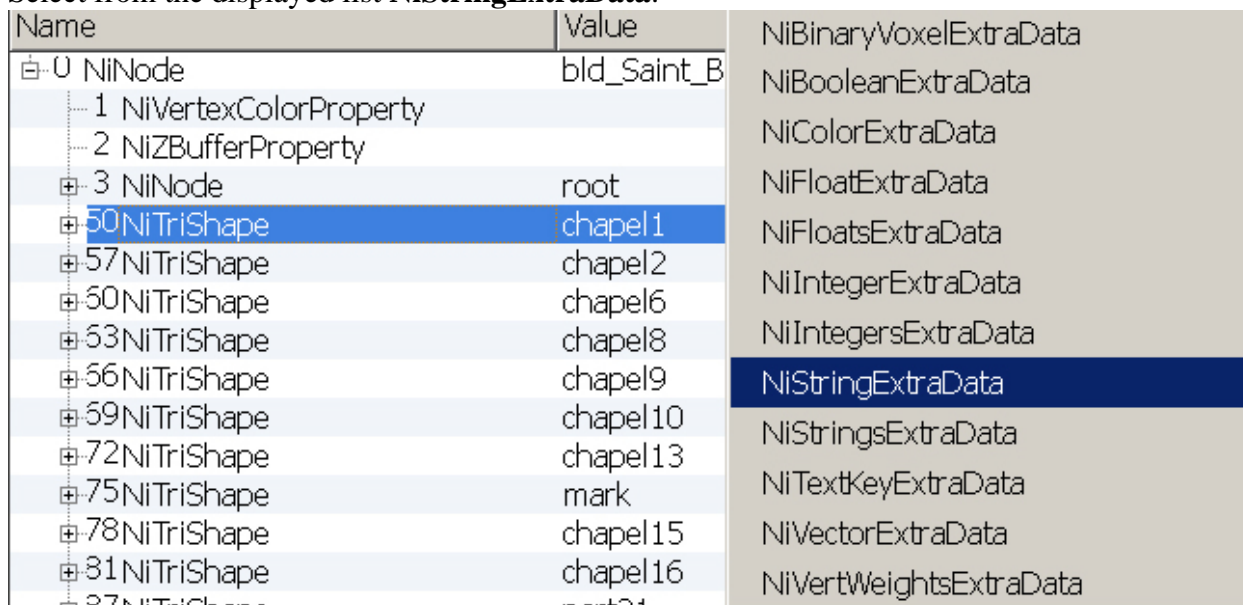
We are interested in the tree of forms (**NiTriShape**). We'll have to edit the shapes we want the snow to fall on.

Right-click the desired shape (**NiTriShape**) and select **Node** → **Attach Extra Data**:





Select from the displayed list **NiStringExtraData**:



A selection of additional information we just added  
(**NiStringExtraData**):



and in the tab **Name** write «**UserPropBuffer**» (no quotes).

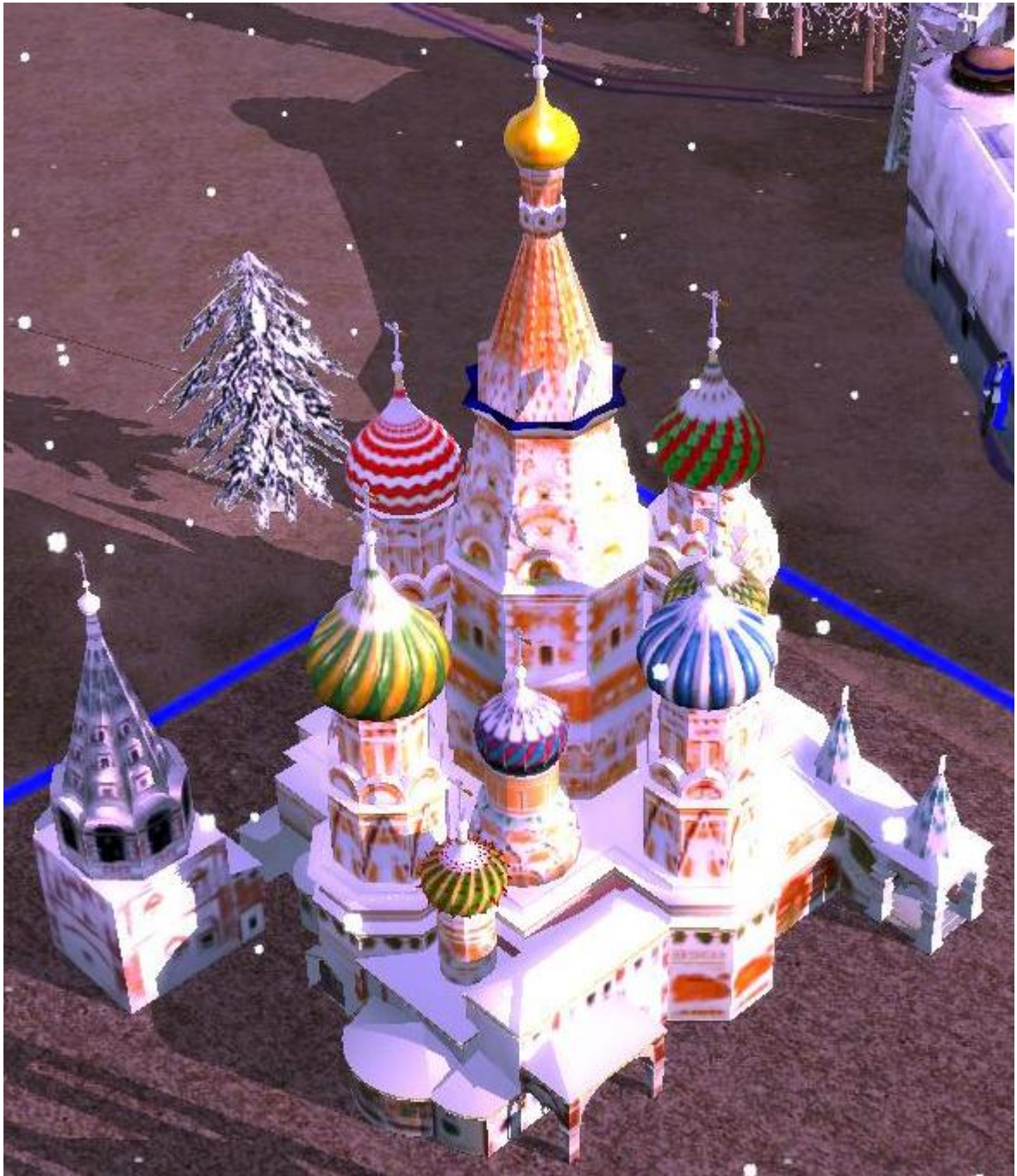
And in the tab **String Data** – «**AnimatedTextureSetName** = <**SaintBasilsCathedral**>».

Name	Type	Value
Name	string	UserPropBuffer
Next Extra Data	Ref<NiExtr...	None
Bytes Remaining	uint	0
String Data	string	AnimatedTextureSetName = <SaintBasilsCathedral>

After performing these actions, we save the changes with all the necessary frames.

For the winter version of the texture, you need to create your own signature player bar. You can take the texture from the summer version (**SaintBasilsCathedral\_pc.tga**) and rename it to **SaintBasilsCathedral\_wi\_pc.tga**, done (provided that the winter texture does not differ from the summer texture in width and height, and the colorless area is in the same place).

That's all. Place the textures **SaintBasilsCathedral.tga**, **SaintBasilsCathedral\_pc.tga**, **SaintBasilsCathedral\_wi.tga** and **SaintBasilsCathedral\_wi\_pc.tga** next to each other in the game folder, and our cathedral will have its own distinctive stripes and will be covered with snow during the winter.



## 10 How to add new textures for land cover, mountains, cliffs, river beds and oceans.

There are 3 types of climate in the game: temperate, arid and tropical. The ground textures may change within 12 months, this rule only applies to temperate climate.

The files in which the textures we need are saved are as follows:

DbSeasonalTextureSets\_Arid.csv - arid

DbSeasonalTextureSets\_Temperate.csv - temperate

DbSeasonalTextureSets\_Tropical.csv - tropical

Located in

EE2X\_db.zip\EE2X\_db\Terrain

These files have names such as:

Grass\_Temperate,terr\_grass\_wi\_02.tga      terr\_grass\_wi\_03.tga      terr\_grass\_sp\_01.tga  
terr\_grass\_sp\_02.tga      terr\_grass\_sp\_03.tga      terr\_grass\_sp\_04.tga      terr\_grass\_su.tga  
terr\_grass\_su.tga      terr\_grass\_fa\_01.tga      terr\_grass\_fa\_02.tga      terr\_grass\_fa\_03.tga  
terr\_grass\_wi\_01.tga

Grass\_Temperate – the name of the texture, by the name you can understand what it is responsible for, in this case - for the grassy cover of the earth in a dry climate. There are many such textures that are responsible for a specific type of earth's surface - up to 6-8 pieces. The game brings them together, resulting in a randomly generated overall texture of the ground - making it look more realistic.

Then there are 12 textures in a row, they are responsible for 12 months, respectively. The countdown starts in January (terr\_grass\_wi\_02.tga).

In these files, you can find textures responsible for land cover, mountains, coastlines, cliffs, river beds and oceans. You can replace any of these textures. To do this, your texture must be converted to tga format and must be 256 by 256 pixels in size. The weight of the texture is also important, it should be 193 kb. This is easy to do if you use Nero PhotoSnap texture (included in Nero version 7) for conversion. Nero PhotoSnap holds tga textures of 256 by 256 pixels, they are always 193 kb in size.

After creating the required texture, rename it. It should be named after the original texture you intend to replace (for example, terr\_grass\_wi\_02.tga). Then put that texture in your zip folder and when you start the game it will load the texture first.

Here is a rough list of the textures responsible for a certain proportion of the land cover:

Bottom of the ocean

Tropical:      terr\_beach\_oceanfloor.tga

Temperate:      terr\_coral.tga

Arid:      terr\_beach\_oceanfloor.tga

Grass

#### Tropical:

terr\_trop\_clay\_shrubs.tga  
terr\_trop\_cover.tga  
terr\_trop\_cover\_rocky.tga  
terr\_trop\_dense\_grass.tga  
terr\_trop\_grass.tga  
terr\_trop\_grass\_patchy.tga  
terr\_trop\_kudzu.tga  
terr\_trop\_mossy\_cover.tga  
terr\_trop\_understory\_cover.tga

#### Desert:

terr\_arid\_dense\_grass.tga  
terr\_arid\_desert\_scrub.tga  
terr\_arid\_grass.tga  
terr\_arid\_mountain.tga  
terr\_arid\_mountain\_grass.tga  
terr\_arid\_small\_scrub.tga  
terr\_arid\_sparse\_grass.tga

#### Mountains

##### Desert:

terr\_arid\_mountain.tga  
terr\_arid\_mountain\_grass.tga

##### Tropical:

terr\_trop\_mountain.tga

##### Temperate:

terr\_mountain01\_su.tga  
terr\_mountain\_top\_wi\_01.tga  
terr\_mountain\_top\_wi\_02.tga  
terr\_mountain\_top\_wi\_03.tga  
with wi particle - winter version of the texture

#### Cliffs

Tropical     terr\_trop\_cliff01.tga  
Arid         terr\_arid\_cliff01.tga  
Temperate     terr\_cliff01.tga

#### Rivers and coasts:

terr\_riverbank\_su.tga  
terr\_riverbottom.tga

My experience with texture change:







## 11 How to add new introductory videos and change videos in the menu. How to change the menu background and loading screens.

All in-game videos are in bik format and can be found in the movies and movies\_ee2x folders. This is a Bink Video format specifically designed for games. You can play it with RAD Video Tools. You can download this program here:

<http://www.radgametools.com/bnkmain.htm> This program also allows you to convert video files in the most popular formats to bik format.

After starting the game, you will see 5 introductory videos. These movies are:

SierraLogo.bik

Nvidia.bik

Intel.bik

md\_intro.bik

EE2\_E3\_VUG.bik

There is also a video from the main menu, this video has no audio:

fe\_loop\_EE2X.bik (for the supplement) or

fe\_loop.bik (to the original game)

You can replace any of these videos with your own by converting the video to motorcycle with RAD Video Tools and overwriting the original bik file. You can make it easier - find your favorite bik videos in other games and don't convert. I didn't notice any requirements for video resolution, bitrate and duration, so almost any bik file will be appropriate.

Now let's move directly to the menu background and loading screens. Then I will list the graphic files and what they are responsible for. You can make exactly the same files (they are textures in tga or bmp format), name them after the original files and put them in the zips folder. In this case, the game will load the files on startup, not the original ones. How you create backgrounds and loading screens depends on your imagination.

What files are responsible for what:

splash\_EE2X.bmp - 300\*150 pixels

responsible for the small image with the logo that appears for the first time when starting the game E.g. (image sizes in the examples are not respected):



before



after

loading\_scren\_1A.tga - 1024\*512 pixels

loading\_screen\_1B.tga - 1024\*256 pixels

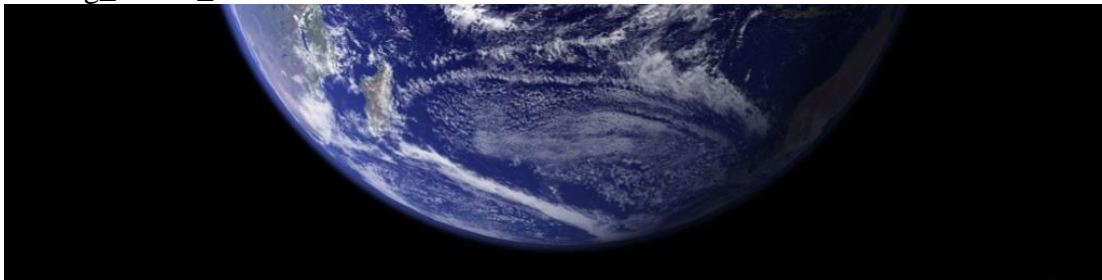
Two files, A - upper half, B - lower half. These files are responsible for the image that appears when running individual scripts:

for example:

loading\_scren\_1A



loading\_screen\_1B



ui\_ingame\_bottom\_right\_nopip\_metal\_2x.tga

512\*512

pixels

ui\_ingame\_bottom\_right\_nopip\_parch\_2x.tga

512\*512

pixels

ui\_ingame\_bottom\_right\_nopip\_stone\_2x.tga

512\*512

pixels

Pictures that appear behind the picture-in-picture position in the lower right corner of the game window. metal - epoch 11-15, scab epoch 6-10, stone - epoch 1-5. Note that the image itself only takes up a quarter of the total texture area:

for example:

ui\_ingame\_bottom\_right\_nopip\_metal\_2x



spr\_EE2X\_logo.tga 1024\*1024 pixels

Loading screen image. It appears immediately after starting the game.

Menu pictures:

These photos are for the background of the menu. Better to make them as texture, without a clear pattern

ui\_fe\_bkg\_parch\_bottom\_left.tga 256\*256 pixels

ui\_fe\_bkg\_parch\_bottom\_mid.tga 256\*256 pixels

ui\_fe\_bkg\_parch\_bottom\_right.tga 256\*256 pixels

ui\_fe\_bkg\_parch\_bottom\_left\_hires\_logo2.tga 512\*512 pixels

ui\_fe\_bkg\_parch\_top\_left\_hires.tga 512\*512 pixels

ui\_fe\_bkg\_parch\_top\_left.tga 256\*256 pixels

ui\_fe\_bkg\_parch\_top\_mid.tga 256\*256 pixels

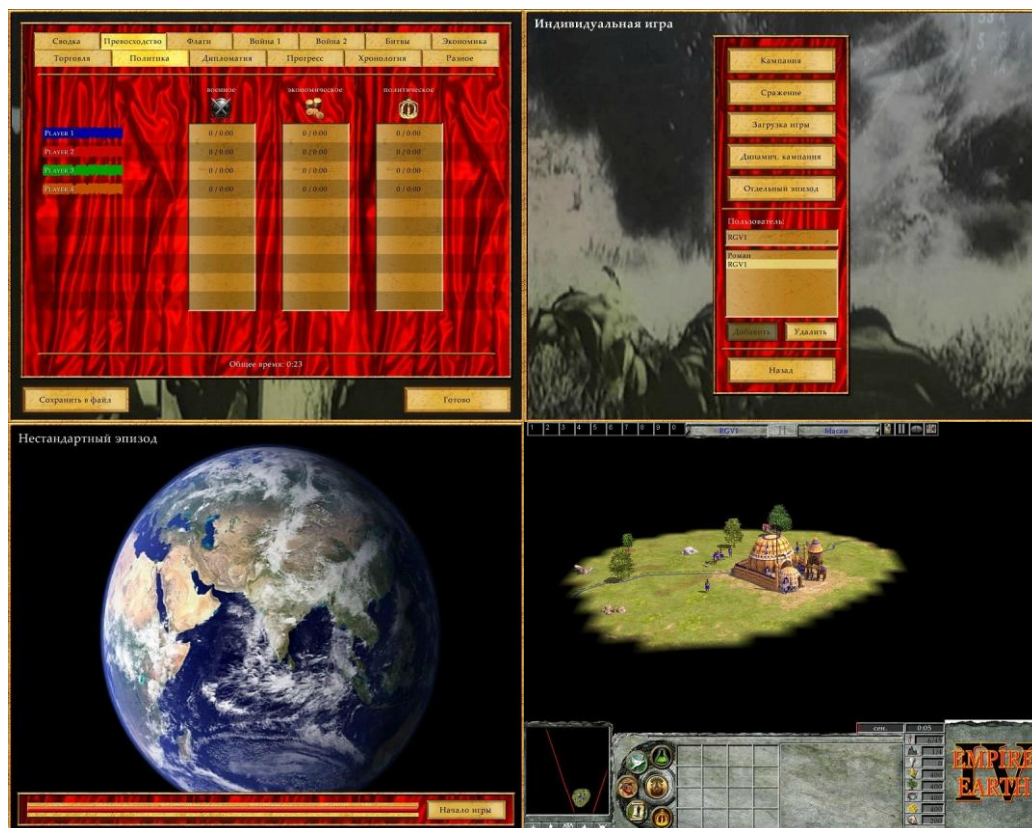
ui\_fe\_bkg\_parch\_top\_right.tga 256\*256 pixels

////////////////////

ui\_fe\_insetpanel\_small\_logo2.tga 256\*256 pixels

This photo is used in the campaign menu when no campaign has been selected yet

Example of changing menus and loading screens:



# 12 Mods for Empire Earth II. Modders Empire Earth II

There are only two major modifications to Empire Earth II.

1 Empire Earth IV Mod – huge mod, the main purpose of its creation is to make each country individual, each era individual, different combat system in each era, to diversify the economic and social element of the game. The mod was released with playable updates. As of this writing, version 9.3 is the latest version of the mod. The official website of the mod, you can download it from here: <http://empireearth4.ucoz.com>

Official mod group vkontakte: [http://vk.com/empire\\_earth\\_4\\_mod](http://vk.com/empire_earth_4_mod)

2 Realistic MOD – great mod, the main goal is to make the original game more realistic. More precisely, more new features, more realistic balance, and much more. As of this writing, 48 is the latest version of the mod. You can download it from here: <http://ee.heavengames.com/downloads/showfile.php?fileid=2925>

You can find many smaller mods on this page:

[http://ee.heavengames.com/downloads/lister.php?category=ee2\\_mods](http://ee.heavengames.com/downloads/lister.php?category=ee2_mods)

There are very few mods for Empire Earth II. There are also very few people who can handle the graphics of the game, they can be counted on one hand.

iwanicki – One of the first to start creating major mods for EE2. Russian.

Michael34 – makes many original small modifications. Spanish.

RGV1 – He was the first to open up the possibility of adding a new 3D model to the game by the creator EE4 Mod.

You can always find them on the site:

[http://ee.heavengames.com/downloads/lister.php?category=ee2\\_mods](http://ee.heavengames.com/downloads/lister.php?category=ee2_mods)

And the RGV1 also has a website <http://empireearth4.ucoz.com> and BK website <http://vk.com/rgv1rgv1>

## Conclusion

Hope you found the information useful. Good luck modding Empire Earth II.

**RGV1**