

Black & White 2

New Miracles



Created By: Bill
Date: February 2 2020

Version	Changes	Date
0	Initial Release	February 2 2020

Table of Contents

Introduction.....	1
Totems.....	2
Power Levels.....	4
Miracles.....	5
Base Miracles.....	6
Miracle Wheat.....	6
Miracle Forest.....	7
Miracle Ore.....	9
Miracle Lava.....	11
Miracle Animals.....	12
Miracle Lights.....	14
Coding.....	15
Constants.....	15
Interface Scripts.....	16
Creating a Totem.....	16
Adding Miracles.....	17
Starting the New Miracles Mod.....	17

Introduction

In the original Black and White game miracles were a prominent and immense feature, with numerous types of miracles that can be used for: gaining resources, aggression and defensive purposes, even temporary creature modifications. The miracle mechanic of the sequel game Black and White 2 (BW2) however is considerably simplistic in comparison. It is comprised of only six miracles: water, heal, shield (wind), lightening, fire, and meteor.

Another significant difference between the games, is how the player comes into possession of new miracles. In the original game, towns bear their own set of miracles. The player obtains access to a town's miracle after convincing them to believe in him. In comparison, BW2 miracles are purchased through the toolbar, as long as the player has the necessary amount of tribute. The mechanics of the original game forces the player to interact with the other towns, as some towns will provide a better strategic advantage due to the miracles they provide the player. Comparatively, the towns in BW2 serve to provide access to more resources, or strategic military positions.

There are further differences in miracle mechanics regarding the produce of mana. In the original, after capturing a town a worship site would appear at the temple. The site would display all the miracles belonging to that town and provide a place for the villagers to worship, generating mana; The player was also expected to care for the worshipers, provide them food, and the occasional heal. Miracles could be activated by clicking on them at the worship site, as long as there is enough mana to generate the miracle.

Contrast that with BW2, where there is an alter building that can be built anywhere, as many times as the player wants. To generate mana the player must create disciple worshipers at these alters, and unlike the original game the worshipers have no desires or need for the player's care. All the alters contribute to the player's overall mana, rather than each worship site having it's own collection of mana like the original game. This sort of mechanic only exacerbates the issue of the player's ability to confine himself to a single town.

The project, referred to as New Miracles (NMs), was first conceived originally in 2014 to expand upon the miracle mechanics of BW2. This initial conception saw the addition of miracles to the alter, as seen in the BW2 expansion pack Battle of the Gods. The 2014 NMs version was bulky and exceptionally buggy. Certainly not something worthy of being posted publicly. Adding miracles to the alter is extremely limited, as there is a fix number of spots due to the design of the alter. The other issue with this first version was in the structuring of the code, as it made it incredibly difficult to program new miracles.

It wasn't until 2019 that the NMs version you are now reading of (code name: Totems) was designed. The requirements of the final product were:

1. Reliable – Little to zero bugs.
2. User Friendly – The controls for the player of the game need to be intuitive and easy to understand.
3. Transferability – It is important that others can easily integrate this mod into their own custom maps.

4. Easy Multiplication – Adding new miracles later on is simple and at least somewhat intuitive, requiring some knowledge of scripting.
5. Strategic – Adds interesting elements to the game.

To fulfill the requirements of (4) and (5) it was decided that adding miracles to the alter would not suffice, as such a plan would add very little other than the miracles themselves to the strategic elements of the game. Moreover it complicated the problem of how the player was to come into possession of the new miracles given the limitations of the scripting language. Thus it was determined that a new building would have to be added via scripting, this building is referred to as a Totem, borrowing the name from the original game.

Totems

A Totem serves as the player's interface into this NMs mod. Each Totem will be connected to a town, and the miracles will appear around it. The miracle bubbles will act similarly to the alter to ensure the fulfillment of requirement (2), since activation of a miracle occurs by simply clicking its symbolic representation at the Totem, exactly as the default BW2 miracles work at the alter.

Given the limitations of BW2 modding at this time, new building cannot be added into the game, thus a combination of other structures and objects were used to represent the Totem. The Totem (Illustration 1) consists of the creature statue base, as seen on the Greek homeland where the villager is praying for assistance, and the town center desire for worship statue.



The miracle displayed above the statue is the town's main miracle, subsequent miracles are added in rows of six around the base of the Totem (Illustration 2). The default code setup allows for 3 rows plus the main, which means there can be 19 miracles per Totem. However that number can be changed easily.

From Illustration 1 and 2, blue flames can be seen in the palms of the Totem's statue. These flames display the Power Level of that Totem. To add more interesting strategic elements it was decided that the effects of miracles be malleable, see section Power Levels for more details.

Illustration 1: Totem with miracle Wheat

Due to limitations of the scripting language the Totems cannot be damaged, therefore the player will not lose access to the miracles unless they run out of mana.

One of the more interesting issues, is how can the player gain access to the miracles of another town. With the Totems, each town's miracles will be on display, it will exist somewhere on the land. Thus all the player needs to do is capture the town, this is simple enough when capturing by force, however migrations present an issue. The solution is for migrations to "carry" their miracles with them to which ever town they are migrating to.

Lets use an example: Town 1, owned by the player as a Totem possessing Miracle Forest. Town 3,

which is neutral has a Totem possessing Miracle Ore. When town 1 gains enough impressiveness to cause town 3 to migrate, the representation of Miracle Ore above town 3's Totem will disappear. The migration will continue as normal, there is no indication of the miracle in the migration. Once the player accepts the migration, a representation of Miracle Ore will appear on town 1's Totem.



Illustration 2: Totem with a full row of miracles



Illustration 3: A Totem with an extreme amount of power

Power Levels

Every Totem has a Power Level, which will modify the effectiveness of its miracles when activated. The Power Level of a Totem is calculated base on the population of the town it is connected to. This was added to complicate the strategic elements, now the player can increase the effectiveness of his miracles by taking care of his town.

The general formula used for calculating a Totem's power is the following:

$$L = \frac{P}{100} \quad [1]$$

Where L is the resulting Power Level of the Totem, and P is the town's population. For example with equation 1 a town with a population of 323 will result in a power level of 3.23.

However an addition modifier is added to [1], based upon the number of miracles the Totem possesses. The reason is the when there is more than 1 miracle the Power Level must be "split" between the multiple miracles. The modifier formula is the following:

$$M = 1 - 0.1(m - 1) \quad [2]$$

Where M is the resulting modifier, and m is the number of unique miracles the Totem contains. The minimum M is allowed to reach is 0.1 which is when $m = 10$. See Table 1.

# of Miracles (m)	Modification Value (M)
1	1
2	0.9
3	0.8
4	0.7
5	0.6
6	0.5
7	0.4
8	0.3
9	0.2
10	0.1
11	0.1

Table 1: Modification Value given number of unique miracles

Therefore the final equation for the cases in which $m \leq 10$ is:

$$L = \frac{P}{100} [1 - 0.1(m - 1)] \quad [3]$$

For all other cases in which $m > 10$, the value of M will be set to 0.1. The modification essentially results in: the more miracles the Totem contains the greater the population needed to obtain higher Power Levels for the miracles. See Illustration 4.

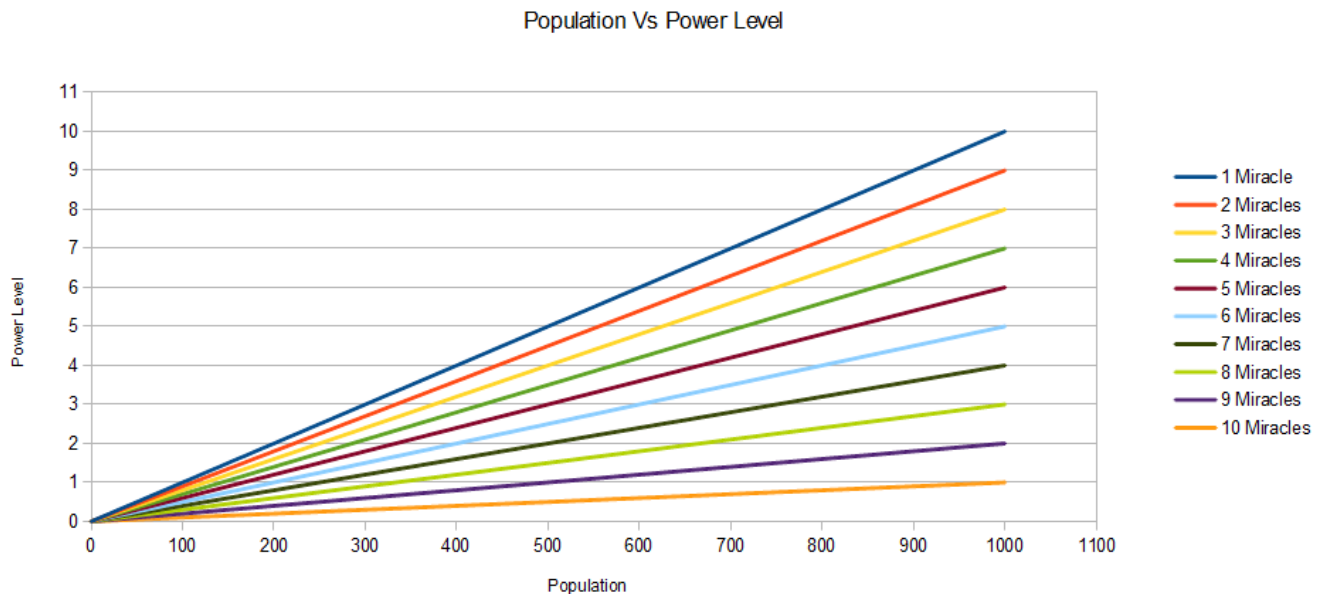


Illustration 4: Resulting Power Levels given town's population and number of miracles

Power Levels are not constantly being calculated, rather once every 30 seconds. It is important to keep this in mind, as in the case in which the player's town experiences a dramatic increase in population it could take up to 30 seconds for the Power Level to be updated.

Finally Power Levels also effect the amount of mana a miracle requires to cast. As the miracle's power increases the amount of mana increases as well. This change in needed mana cannot be characterized with mathematical formulas because every miracle contains its own formula. However it is something to keep in mind when playing a game with this mod installed.

Miracles

The interface for using miracles is similar to the original miracles featured in BW2. By clicking the symbolic bubble of the miracle at the Totem will begin the activation process; the Totem must also be connected to a town belonging to the player.

First the mana required for the miracle will be taken from the alter. If in the case which there is not enough mana to create the miracle a charging visual will appear at the player's hand. As the amount of mana increases the charge visual will grow until the needed amount is reached.

The miracle will appear in the player's hand. There are three actions which the player can take, throw, pour, or cancel.

Throwing and pouring will differ from miracle to miracle. The miracle cannot be activated outside the player's influence. If the player is pouring and reaches outside the influence the pouring will be paused, until the hand returns within the player's influence.

Canceling is done by pressing backspace, delete, or which ever button the player has mapped to the cancel action. Note canceling by shaking one's hand doesn't work.

Base Miracles

The default New Miracles package comes with 6 miracles: wheat, forest, ore, lava, lights, and animals.

Miracle Wheat

The Wheat Miracle magically produces grain. The symbolic bubble of the miracle contains the straw bale (see Illustration 5).

When thrown, the miracle will produce small piles of grain randomly place within 25 units of the contact point (see Illustration 6). However if there happens to be a building near by the gain will be added to it.

When poured, grain will fall from your hand, adding onto any pile of grain or building near it (see Illustration 7).



Illustration 5: Symbolic bubble of Miracle Wheat



Illustration 6: Effect of Miracle Wheat when thrown



Illustration 7: Effect of Miracle Wheat when being poured

As mentioned before the amount of grain produced will change with the town's Power Level. The equation followed by the miracle is:

$$A = 1000 * L \quad \text{Where } A = \text{the amount of gain produced.}$$

For example a power level of 2.5 will produce 2500 units of grain. It should be noted that there is no limit of how much grain will be produced, meaning the player could have a power level of 30 and it would still produce $30 * 1000$ grain units. However there is a minimum which is 1000 units of grain.

The amount of mana required is:

$$M_w = 750L \quad \text{Where } M_w \text{ is the mana required to activate Miracle Wheat.}$$

The minimum amount of mana needed is 750.

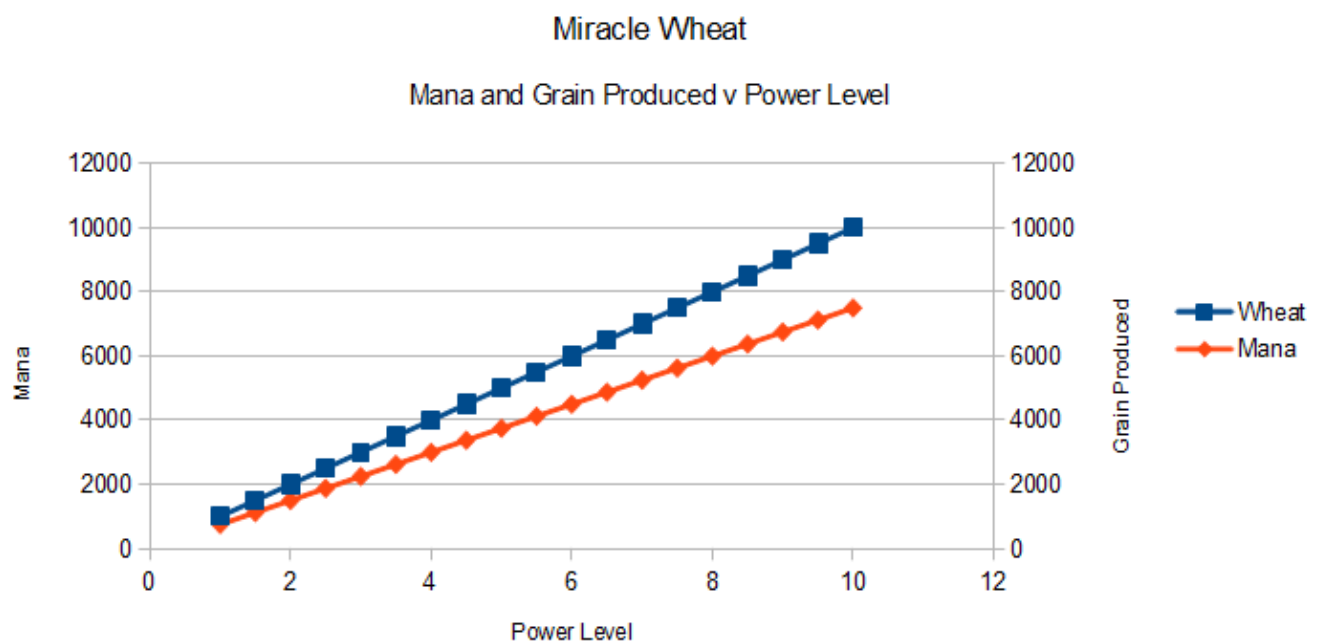


Illustration 8: Miracle Wheat result and mana graphed

Miracle Forest

This miracle was modeled after the Miracle Forest seen in BW1, where upon casting several trees will appear. The symbolic bubble has a tree and green visuals swirling around it (see Illustration 9).

When thrown trees will be flung from the contact point (see Illustration 10), these tree will not be planted into the grown, which could cause damage to near by building or villagers. When poured trees will grow out of the ground around the hand (see Illustration 11), as more trees grow the radius in which they can appear will also grow.



Illustration 9: Symbolic bubble of Miracle Forest



Illustration 10: Miracle Forest when thrown



Illustration 11: Miracle Forest when poured

As the player's power level increases, so does the amount of trees per this formula:

$$A = 2 * L \quad \text{Where } A = \text{the number of trees created.}$$

There is no maximum, but there is a minimum which is 2 trees. The result is rounded upwards. For example, an L of 1.1 will result in 3 trees.

The amount of mana required is:

$$M_f = 850L \quad \text{Where } M_f \text{ is the mana required to activate Miracle Forest.}$$

The minimum amount of mana needed is 850.

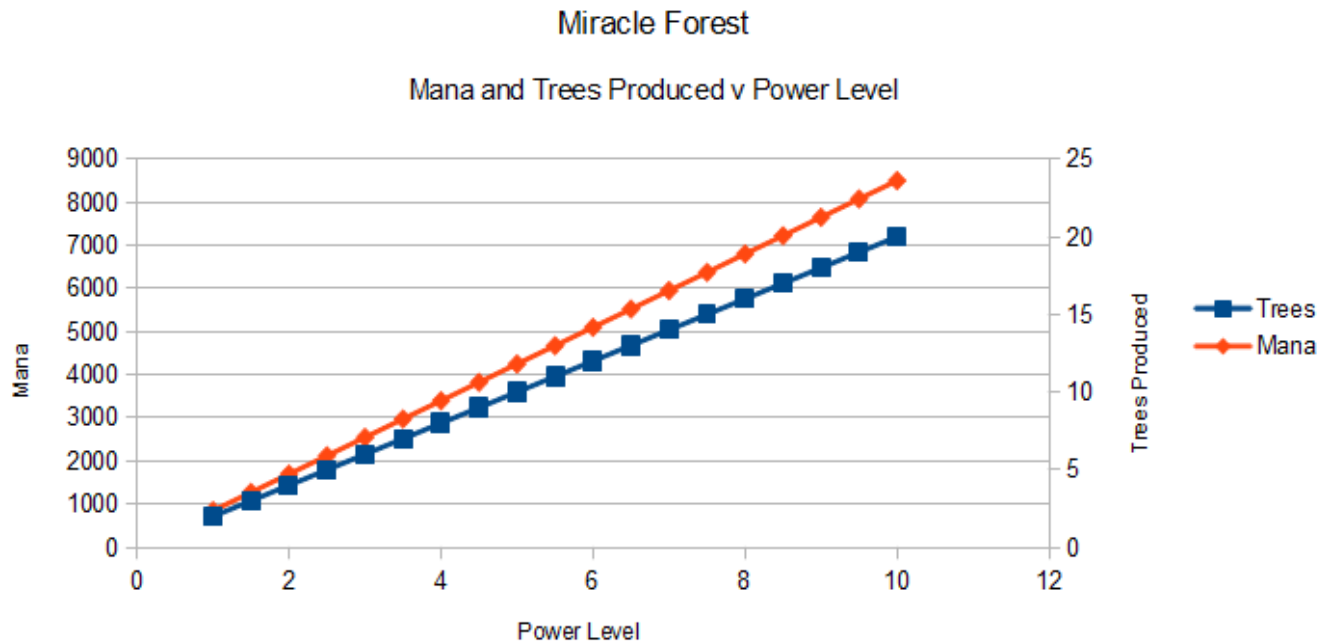


Illustration 12: Miracle Forest result and mana graphed

Miracle Ore

Similar to the previous two, this miracle is designed to provide another way to access a resource, in this case ore. This miracle doesn't produce processed ore, but rather ore rocks that the player can transform into something usable. The symbolic bubble for Miracle Ore contains an ore rock (see Illustration 13).

When thrown the miracle explodes into numerous ore rocks (see Illustration 14), poured will drop ore rocks wherever the hand is located (see Illustration 15).

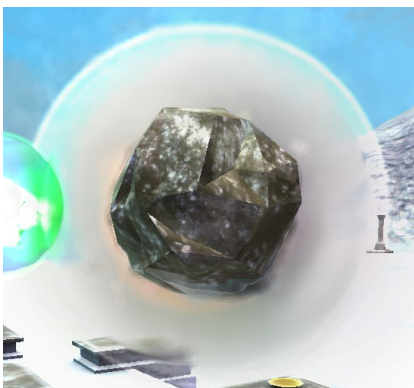


Illustration 13: Symbolic bubble of Miracle Ore



Illustration 14: Miracle Ore when thrown



Illustration 15: Miracle Ore when poured

The number of rocks will change depending on the Totem's Power Level. The variations follow the formula:

$$A = 3 * L \quad \text{Where } A \text{ is the number of ore rocks produced.}$$

There is no maximum amount, but there is a minimum, which is 3 rocks. The size of the ore rocks when create will vary slightly from 0.8 to 1.1.

The amount of mana required follows the formula:

$$M_o = 1000 * L \quad \text{Where } M_o \text{ is the mana required to activate Miracle Ore.}$$

The minimum amount of mana needed is 1000.

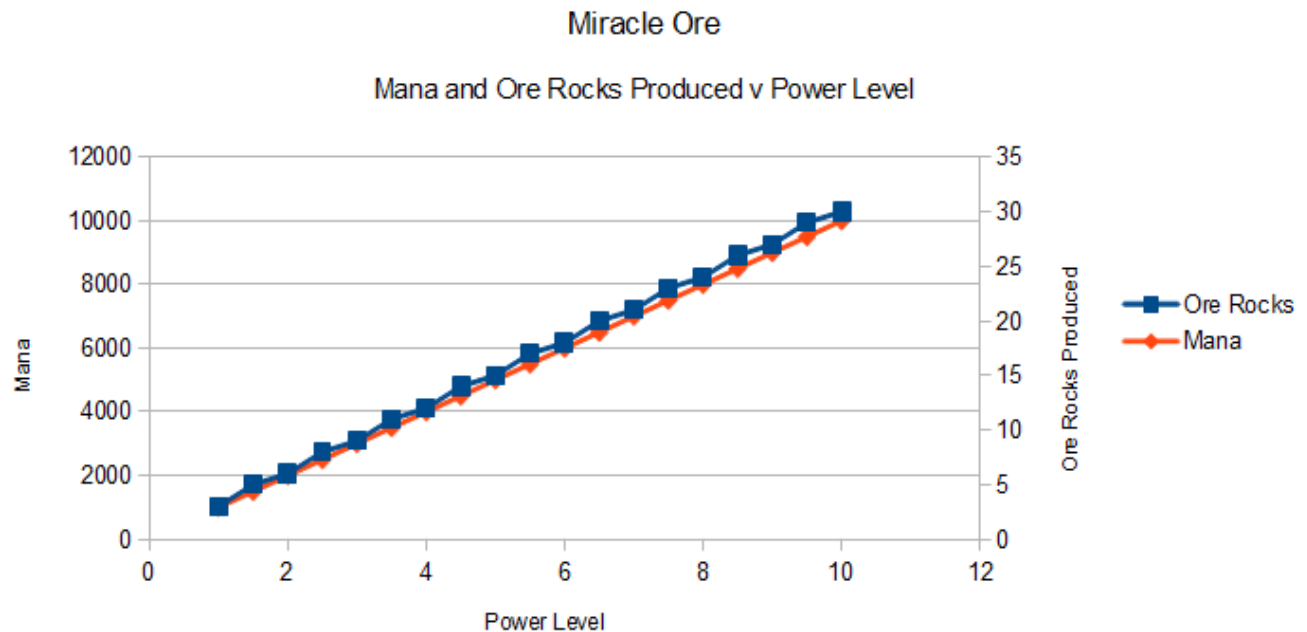


Illustration 16: Miracle ore results mana graphed

Miracle Lava

The only aggressive miracle provided in the base package is Miracle Lava. Modeled after the lava miracle featured in BW2 expansion pack Battle of the Gods. This miracle creates a lava flow which will burn buildings and villagers. See Illustration 18 for the symbolic bubble.

When thrown a lava puddle will appear not only at the contact point but several other puddles will also appear around that point (Illustration 17). The number of the puddles is dependent upon the Power Level.

Pouring is slightly different, as the action button is held down lava puddles will appear beneath the hand, until the maximum number of puddles is reached (Illustration 19). The maximum is calculated based on the Power Level.

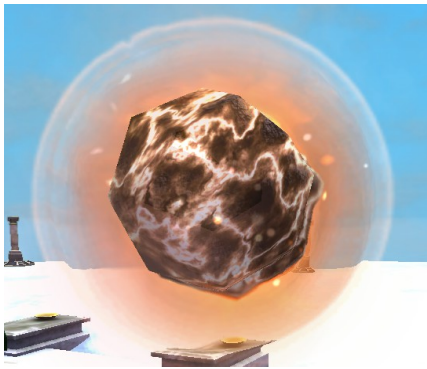


Illustration 18: Symbolic bubble of Miracle Lava

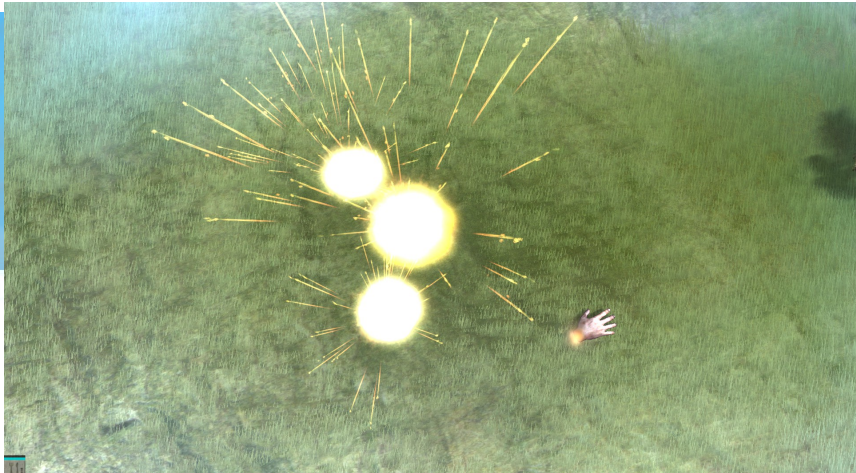


Illustration 17: Miracle Lava when thrown



Illustration 19: Miracle Lava when poured

The number of lava puddles is calculated using the following formula:

$$A = L - 1 \quad \text{Where } A \text{ is the number of lava puddles}$$

There is no maximum number of puddles but there is a minimum, which is 1.

The amount of mana required follows the formula:

$$M_l = 1500 * L \quad \text{Where } M_l \text{ is the mana required to activate Miracle Lava.}$$

The minimum amount of mana needed is 1500.

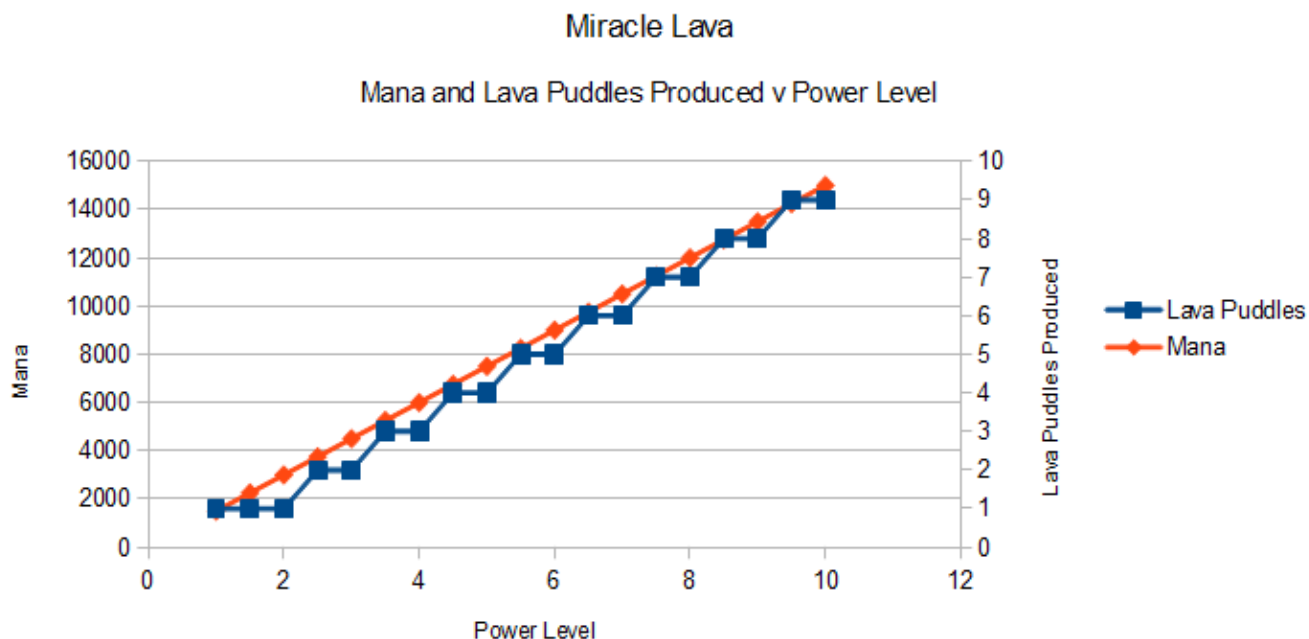


Illustration 20: Miracle lava results and mana graphed

Miracle Animals

In the expansion pack for BW2 there is a miracle called Verdant. This miracle created animals and converted any villagers close by into animals. Miracle Animals is similar in that it creates animals, but it doesn't convert villagers into animals. See Illustration 21 for the symbolic bubble.

When thrown the animals will appear around the contact area; the radius will increase the more animals that are created (see Illustration 22). When poured the animals will appear beneath the hand. The number of animals generated is depended upon the Power Level.



Illustration 21: Symbolic bubble of Miracle Animals



Illustration 22: Miracle Animals when thrown

The number of animals generated is calculated using the following formula:

$$A = 5L \quad \text{Where } A \text{ is the number of animals}$$

There is no maximum number of animals but there is a minimum, which is 5.

The amount of mana required follows the formula:

$$M_a = 500L \quad \text{Where } M_a \text{ is the mana required to activate Miracle Animals.}$$

The minimum amount of mana needed is 500.

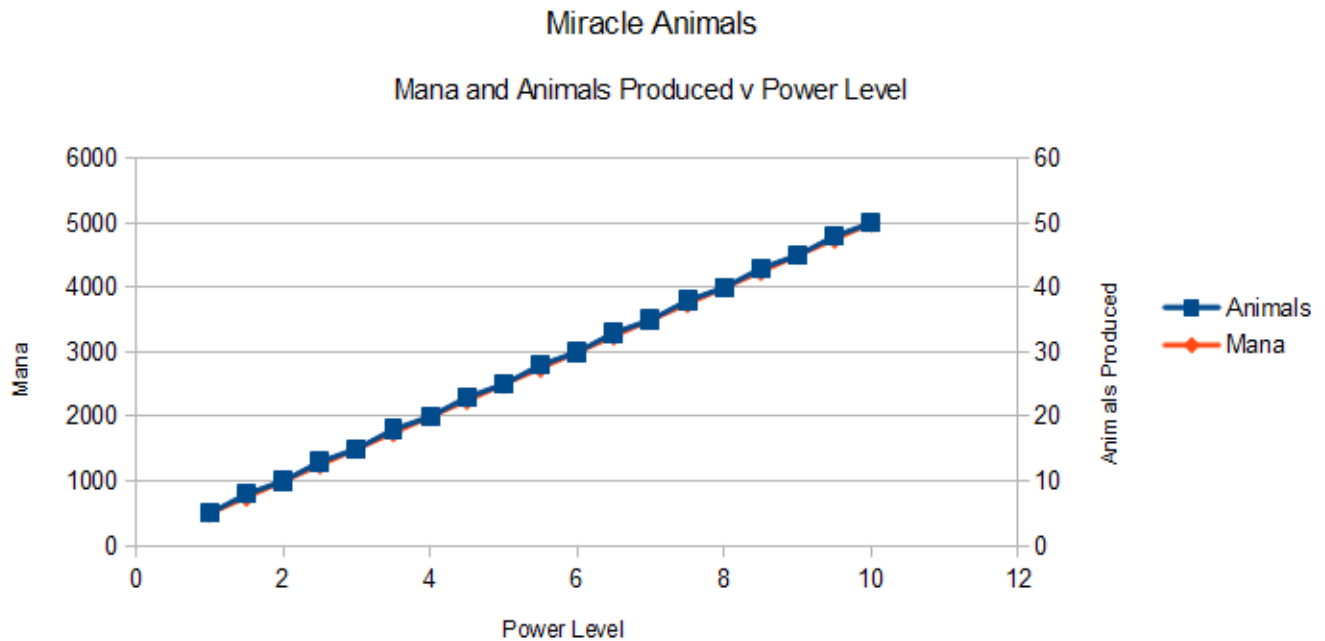


Illustration 23: Miracle Animal results and mana graphed

Miracle Lights

Miracle lights is designed to be a friendly miracle. The miracle impresses the all the villagers near by, as it produces numerous fireworks. The symbolic bubble can be seen in Illustration 24.

When thrown numerous launch points will be generated where fireworks are fired from. Pouring simply fires off fireworks from where the player's hand is. The higher the power level the more fireworks and greater the overall effect.



Illustration 24: Symbolic bubble for Miracle Lights



Illustration 25: Miracle Lights when thrown

The amount of mana required follows the formula:

$$M_g = 750 L \quad \text{Where } M_g \text{ is the mana required to activate Miracle Lights.}$$

The minimum amount of mana needed is 750.

Coding

The New Miracles mod is designed in such a way as to allow others to easily integrate it into their own land. This section of the documentation will not describe the basics of the scripting language, nor how to compile a project, but will discuss how to implement the mod into another land.

The source code of New Miracles is split into multiple files to make understanding the code simpler. The main file is *NMs_Totem.txt*, then each miracle is contained within its own file, with *NMs_Globals.txt* possessing variables and scripts required across all the files. All files, variables, scripts, and constants begin with the prefix “NMs_” which is short for “New Miracles.” This is done to ensure that when added into an ongoing project the names used will not conflict.

Starting with the challenge file of a project, where the mod files must be listed. The order is quite simple. *NMs_Globals.txt* contains variables, constants, and scripts needed across all the miracles and the Totems it must be listed first. *NMs_Totem.txt* by contrast must access all the variables and scripts for each miracle, as such must be listed after all the miracles. Therefore the order of the files must be: (1) *NMs_Globals.txt*; (2) The miracles in no particular order; and (3) *NMs_Totem.txt*.

```
NMs_Globals.txt
NMs_Wheat.txt
NMs_Animal.txt
NMs_Forest.txt
NMs_Ore.txt
NMs_Light.txt
NMs_Lava.txt
NMs_Totem.txt
MainScript.txt
```

Which ever file which will contain the script used for Totem initialization must be listed after *NMs_Totem.txt*, such that it's scripts can be accessed.

Constants

Based on the requirements of the land, some miracles maybe discarded and/or others removed. If this is the case there are a number of constants that will need to be updated. This section will not cover the details of adding a new miracle into the code, but rather assume that the miracle has already been integrated. To learn more about how to create your own miracle see the “Custom Miracle Creation” document.

NMs_NUMMIRACLES – Defines the number of miracles that will exist on the land, generally its one per Totem, however if in the case that a Totem requires more than one miracle then those must be accounted for also. If this value is changed the constant *NMs_MIRDEF* must be updated. The value of this constant is:

$$NMs_MIRDEF = NMs_NUMMIRACLES \times NMs_MIR2D$$

It is important to note that the value of *NMs_MIR2D* must never be changed.

NMs_MIRACLETYPES – This value doesn't need to be edited unless a miracle has been removed or added. If this value is changed then the constant *NMs_TOTEMDEF2* in Totem.txt must also be updated using this formula:

$$NMs_TOTEMDEF2 = NMs_NUMBERTOTEM \times NMs_MIRACLETYPES$$

Also found in *NMs_Global.txt* is the list of miracle type constants which be used to address specific miracles when creating Totems or adding miracles. There are also launching constants, which are used in the forest and ore miracles. Don't change them unless you know what you're doing.

In *NMs_Totem.txt* there is one constant of importance:

NMs_NUMBERTOTEM – The number of Totems that will exist on the land. If changed two other constants need to be updated, (1) *NMs_TOTEMDEF* and (2) *NMs_TOTEMDEF2* the formulas are:

$$NMs_TOTEMDEF = NMs_NUMBERTOTEM \times NMs_TOT2D$$

$$NMs_TOTEMDEF2 = NMs_NUMBERTOTEM \times NMs_MIRACLETYPES$$

Interface Scripts

In the *NMs_Totem.txt* file, there are two types of scripts, static scripts and interface scripts. When adding the mod to a project the interface scripts are the most important. The following is a list of interface scripts:

- *NMs_totemMain*
- *NMs_createTotem*
- *NMs_addMiracleToTotem*

Creating a Totem

To create a Totem the script *NMs_createTotem* must be used. The script definition is:

```
define script NMs_createTotem(xPos,zPos,ang,townId,idx)
```

Parameter descriptions:

- *xPos* – X co-ordinate of the Totem.
- *zPos* – Z co-ordinate of the Totem.
- *ang* – Angle of the Totem.
- *townId* – Id of the town the Totem is connected to.
- *idx* – Main miracle type.

An example of the script being run:

```
run script NMs_createTotem(479,920,220,0,NMs_WHEAT)
```

NMs_WHEAT is a constant in *NMs_Globals.txt*. This is also where a full listing of the miracles and their values can be found.

Each Totem when created is assigned an index in the order they are made starting at zero. Thus the first

Totem will have the id 0, the second 1, third 2, and so on. This is important to understand with adding additional miracles to them.

Adding Miracles

In the case in which more miracles need to be added to a Totem the script *NMs_addMiracleToTotem* can be used. The script definition is:

```
define script NMs_addMiracleToTotem(idx,miracleIdx,m_idx)
```

Parameter descriptions:

- idx – Index of the Totem the miracle is to be added to.
- miracleIdx – Index of the miracle to be added.
- m_idx – Must always be -1.

An example of the add miracle script being used:

```
run script NMs_addMiracleToTotem(0,NMs_LAVA,-1)
```

The execution of the script will take a few moments.

Starting the New Miracles Mod

After all the Totems are created, to start the New Miracle's mod the script *NMs_totemMain* must be started in the background. There are no parameters so the code to start this script is simple:

```
run background script NMs_totemMain
```

Here is an example of all the scripts being used to fill a land with Totems and miracles:

```
run script NMs_createTotem(479,920,220,0,NMs_WHEAT)
run script NMs_createTotem(2126,2006,60,1,NMs_ANIMALS)
run script NMs_createTotem(1526,789,30,2,NMs_LIGHT)
run script NMs_createTotem(707,1598,15,3,NMs_LAVA)
run script NMs_createTotem(1197,2156,90,4,NMs_FOREST)
run script NMs_createTotem(1641,1462,34,5,NMs_ORE)

run script NMs_addMiracleToTotem(0,NMs_ANIMALS,-1)
run script NMs_addMiracleToTotem(1,NMs_FOREST,-1)
run script NMs_addMiracleToTotem(2,NMs_ORE,-1)
run script NMs_addMiracleToTotem(3,NMs_LAVA,-1)
run script NMs_addMiracleToTotem(4,NMs_LIGHT,-1)

run background script NMs_totemMain
```