

Particle Tool

by Atlas and ImBee

Instrukcja

Czym jest Particle Tool By Atlas and ImBee i do czego służy?

Particle Tool By Atlas and ImBee to pakiet dwóch programów służących do konwersji plików particlesów z gier **Reality Pump** w celu umożliwienia ich prostszej edycji i uruchamiania tego samego particlesa w różnych grach tego studia. Pakiet tych programów jest “mostem” pomiędzy różnymi formatami particlesów. Programy te obsługują następujące gry/tytuły:

- Earth 2150: Escape from the Blue Planet,
- Earth 2150: The Moon Project,
- Earth 2150: Lost Souls,
- World War III: Black Gold,
- Heli Heroes,
- Frontline Attack: War over Europe/World War II: Panzer Claws II,
- Polanie II/KnightShift/Once Upon a Knight,
- Polanie III/KnightShift II Curse of Souls
- Earth 2160,
- 3D ParticleGen Visual FX (steam),
- Two Worlds,

Pakiet jest złożony z dwóch programów:

- Particle2MyAod.exe
- MyAod2Particle.exe

Particle2MyAod:

Particle2MyAod.exe to program, który służy do eksportu danych z particlesów w formacie **.msh** i **.prt** z tytułów przedstawionych wyżej do formatu **.myaod**, który jest bardzo podobny do **.aod**. Wystarczy wprowadzić nazwę pliku particlesa wraz z formatem.

Plik z ParticleEmitemerem:

W przypadku wykrycia ParticleEmitera program zapyta o wymuszenie końcowego formatu. Możemy wybrać **y/n**. Jeśli wybierzemy **n** to program wypakuje plik particlesa do **.myaod** nie wymuszając jakiegось konkretnego formatu czyli program po prostu wypakuje particlesa. W przypadku gdy wprowadzimy **y** to będziemy mogli wymusić format do którego później ma zostać skompilowany **.myaod** (np. jeśli chcę uruchomić particlesa z **KnightShift** w **Earth 2160** to wpisuje **y**, a potem wymuszam format **e2160**. Program **MyAod2Particle** już sam będzie wtedy wiedział jak skompilować **.myaod** do formatu **Earth 2160**). Następnie program wypakuje plik particlesa.

W folderze wyeksportowanego particlesa znajduje się plik **.myaod** i plik **_extra_data.cpp**. W pliku **extra_data** znajduje się wypakowany cały początek pliku particlesa w postaci podanych wartości zmiennych. Można to modyfikować ale nie trzeba. Najważniejszy jest plik **.myaod**.

Program działa również w trybie **argc&argv**, więc można go wywołać z **cmd** albo z **powershella** i wpisać argument obok nazwy programu co spowoduje samo wyeksportowanie particlesa bez wymuszenia formatu:

Particle2MyAod.exe <nazwa_pliku_input>

Można również wymusić odpowiedni format następującym sposobem:

Particle2MyAod.exe <nazwa_pliku_input> -force <ks/tw/pg/e2160>

lub

Particle2MyAod.exe <nazwa_pliku_input> --force <ks/tw/pg/e2160>

Plik z Dynamic particlesem:

Obsługa **Dynamic particlesów** niestety odbywa się przez edycję pliku konfiguracyjnego **DynamicParticle.cfg**. W zależności od tytułu, z którego pochodzą particlesy należy zmodyfikować ten plik konfiguracyjny – najlepiej natychmiastowo. Standardowo ustawione są następujące opcje, które znaczą:

dynamic_particle_input_format = ks;

(Wejściowy Dynamic particle będzie pochodzić z KnightShift.)

force_specific_export_format = false;

(Czy wymusić specyficzny format?)

forced_export_format = ks;

(Wyjściowy końcowy format Dynamic particlesa będzie formatem z gry KnightShift)

Aby obsłużyć specyficzny format z innej gry gdzie jest **Dynamic particles** wystarczy wpisać jeden z tych formatów: **e2150**, **ww3_or_hh**, **ww2_or_fa_or_pc2**, **ks**

w to miejsce po znaku =

dynamic_particle_input_format = <tutaj wpisujemy format>;

Aby konwertować do innej gry należy odblokować przełącznik ***force_specific_export_format*** na ***true***

Końcowy format do którego konwertujemy należy wpisać w ***forced_export_format*** po znaku =
forced_export_format = <tutaj wpisujemy format>;

Kompilator już sam powinien wiedzieć jak skompilować katalog z wyeksportowanym particlesem.

Do kompilacji formatu **.myaod** stworzyłem swój specjalny kompilator **MyAod2Particle.exe**.

Aby automatycznie wypakować wszystkie pliki bez wymuszania formatu, można użyć **_EXPORT_ALL.ps1** lub **_EXPORT_ALL.bat**.

MyAod2Particle:

MyAod2Particle.exe to program, który służy do kompilowania plików **.myaod** particlesów do ustalonego wcześniej formatu z danej gry **Reality Pump**.

Program się włącza przez dwuklik i wpisuje się nazwę katalogu wejściowego. Jeżeli plik **extra_data** w katalogu, nie istnieje to program stworzy klasyczny nagłówek particlesa generując odpowiednie informacje, jednak jeżeli istnieje plik **extra_data** to przekompiluje ten plik na nagłówek particlesa.

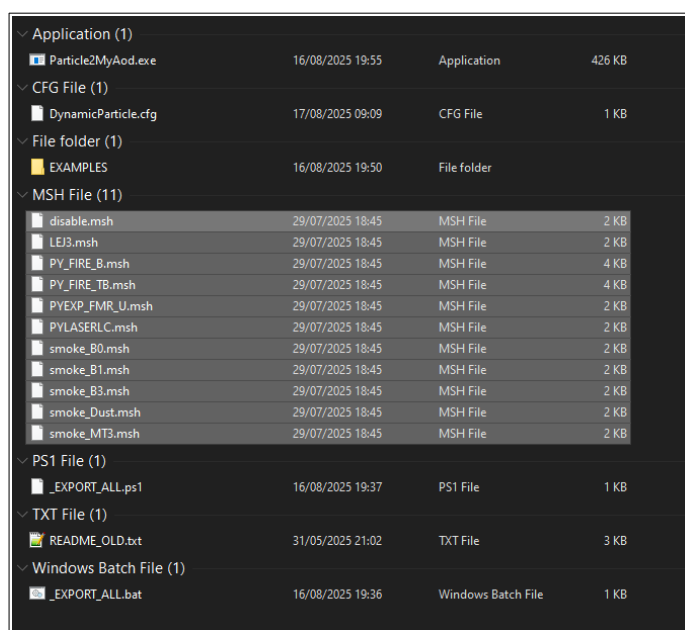
Program działa również w trybie **argc&argv**, więc można go wywołać z **cmd** albo z **powershella** i wpisać argument obok nazwy programu czyli

MyAod2Particle.exe <nazwa katalogu wejściowego>

Aby automatycznie skompilować/zaimportować wszystkie foldery, można użyć **_IMPORT_ALL.ps1** lub **_IMPORT_ALL.bat**.

Typowe i przykładowe scenariusze używania ParticleToola:

a) Chcę wyeksportować Dynamic particlesy z **Earth 2150**:



Application (1)			
Particle2MyAod.exe	16/08/2025 19:55	Application	426 KB
CFG File (1)			
DynamicParticle.cfg	17/08/2025 09:09	CFG File	1 KB
File folder (1)			
EXAMPLES	16/08/2025 19:50	File folder	
MSH File (11)			
disable.msh	29/07/2025 18:45	MSH File	2 KB
LEJ3.msh	29/07/2025 18:45	MSH File	2 KB
PV_FIRE_B.msh	29/07/2025 18:45	MSH File	4 KB
PV_FIRE_TB.msh	29/07/2025 18:45	MSH File	4 KB
PVEXP_FMR_U.msh	29/07/2025 18:45	MSH File	2 KB
PVLASERLC.msh	29/07/2025 18:45	MSH File	2 KB
smoke_B0.msh	29/07/2025 18:45	MSH File	2 KB
smoke_B1.msh	29/07/2025 18:45	MSH File	2 KB
smoke_B3.msh	29/07/2025 18:45	MSH File	2 KB
smoke_Dust.msh	29/07/2025 18:45	MSH File	2 KB
smoke_MT3.msh	29/07/2025 18:45	MSH File	2 KB
PS1 File (1)			
_EXPORT_ALL.ps1	16/08/2025 19:37	PS1 File	1 KB
TXT File (1)			
README_OLD.txt	31/05/2025 21:02	TXT File	3 KB
Windows Batch File (1)			
_EXPORT_ALL.bat	16/08/2025 19:36	Windows Batch File	1 KB

Figure 1: Przykładowo mam jakąś pulę particlesów z **E2150**.

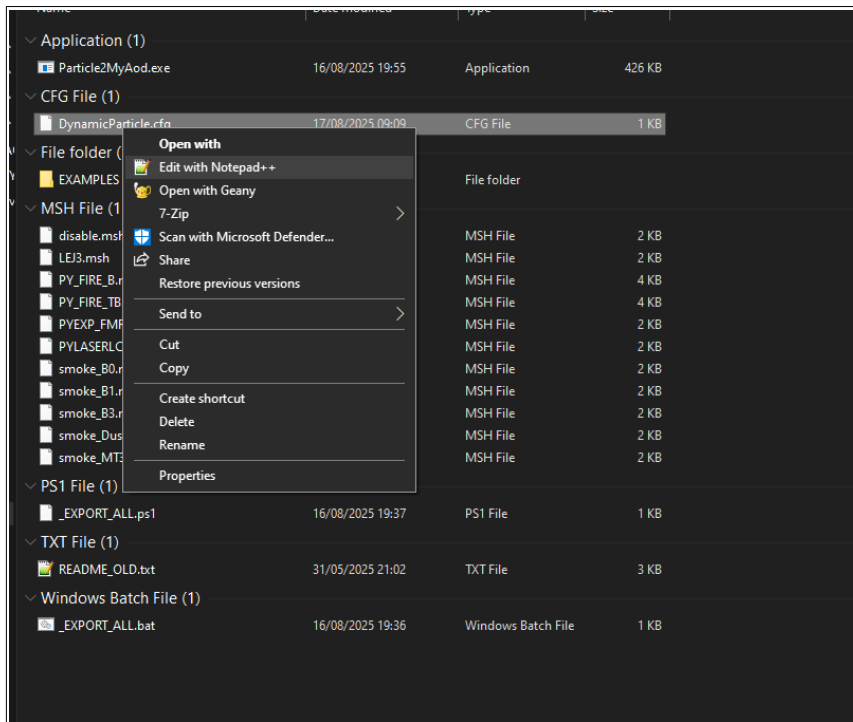


Figure 2: Edytuję *DynamicParticle.cfg*.

```
// Dynamic Particle Config:
dynamic_particle_input_format = ks;
force_specific_export_format = false;
forced_export_format = ks;
// Available formats:
// e2150
// ww3_or_hh
// ww2_or_fa_or_pc2
// ks
```

Figure 3: Zauważam opcje.

```
// Dynamic Particle Config:
dynamic_particle_input_format = e2150;
force_specific_export_format = false;
forced_export_format = e2150;
// Available formats:
// e2150
// ww3_or_hh
// ww2_or_fa_or_pc2
// ks
```

Figure 4: Zmieniam format na odpowiedni wejściowy format.

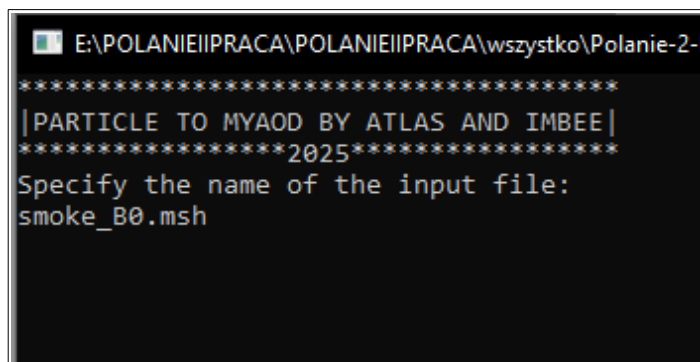


Figure 5: Włączam program i wpisuję nazwę pliku.
Następnie klikam enter.

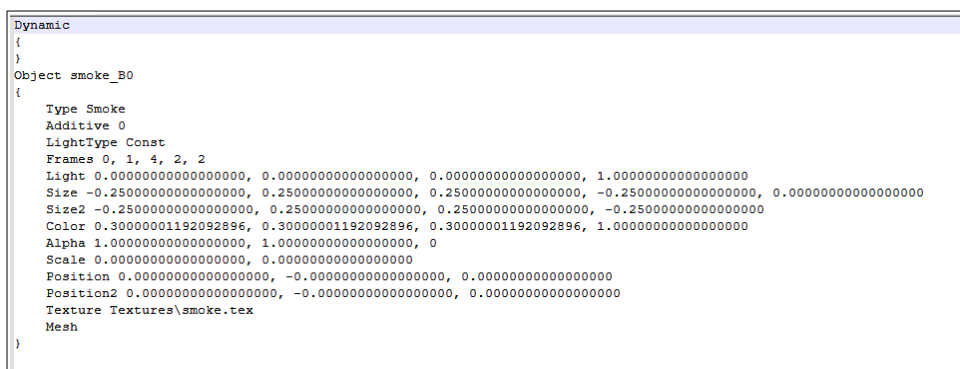


Figure 6: Gotowe - mogę edytować dane Dynamic particlesa.

b) Chcę przekonwertować Dynamic particlesy z Earth 2150 do World War III Black Gold:

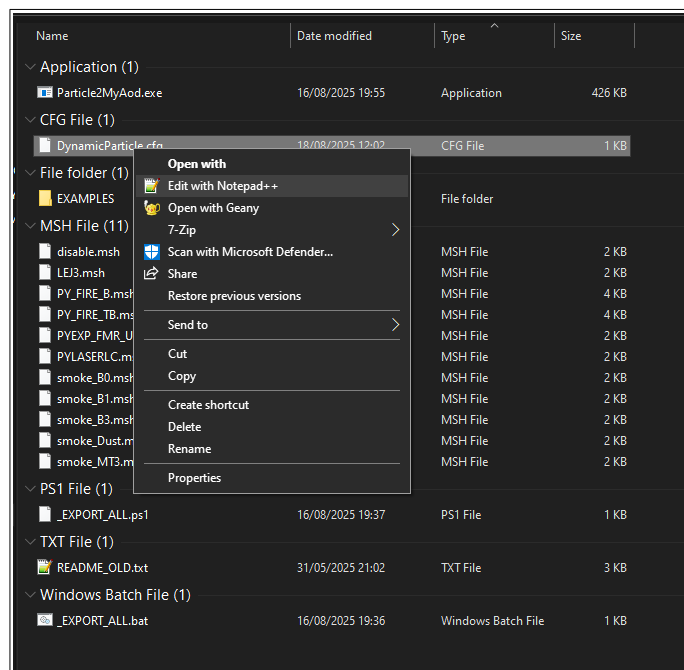


Figure 7: Edytuję DynamicParticle.cfg.

```
// Dynamic Particle Config:
dynamic_particle_input_format = e2150;
force_specific_export_format = false;
forced_export_format = e2150;
// Available formats:
// e2150
// ww3_or_hh
// ww2_or_fa_or_pc2
// ks
```

Figure 8: Zauważam opcje.

```
// Dynamic Particle Config:
dynamic_particle_input_format = e2150;
force_specific_export_format = true;
forced_export_format = ww3_or_hh;
// Available formats:
// e2150
// ww3_or_hh
// ww2_or_fa_or_pc2
// ks
```

Figure 9: Ustawiam format wejściowy na e2150, odblokowuję force_specific_export_format na true i ustawiam forced_export_format na ww3_or_hh.

```
E:\POLANIE\PRACA\POLANIE\PRACA\wszystko\Polanie-2-Da
*****
|PARTICLE TO MYAOD BY ATLAS AND IMBEE|
*****2025*****
Specify the name of the input file:
LEJ3.msh
```

Figure 10: Włączam program i wprowadzam nazwę particlesa i klikam enter.

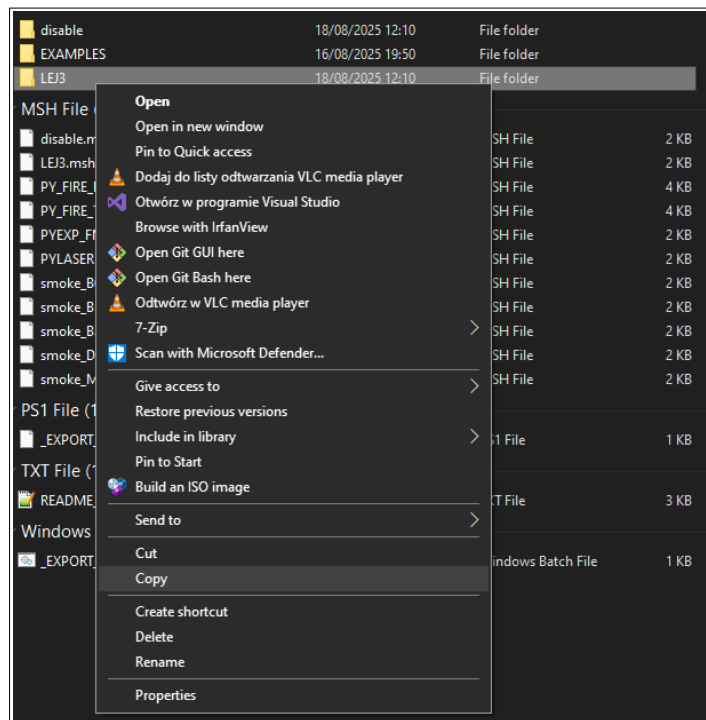


Figure 11: Kopiuję otrzymany katalog z wyeksportowanym particlesem do katalogu z MyAod2Particle.

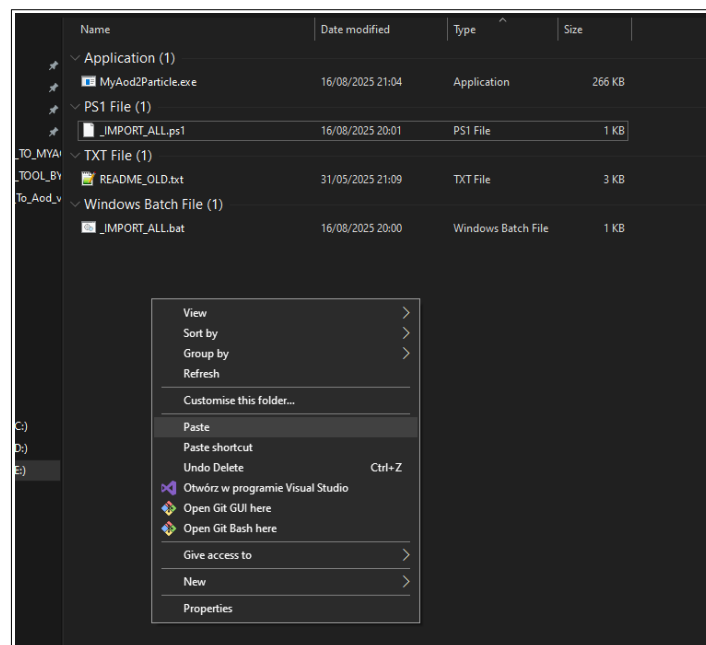


Figure 12: Wklejam.

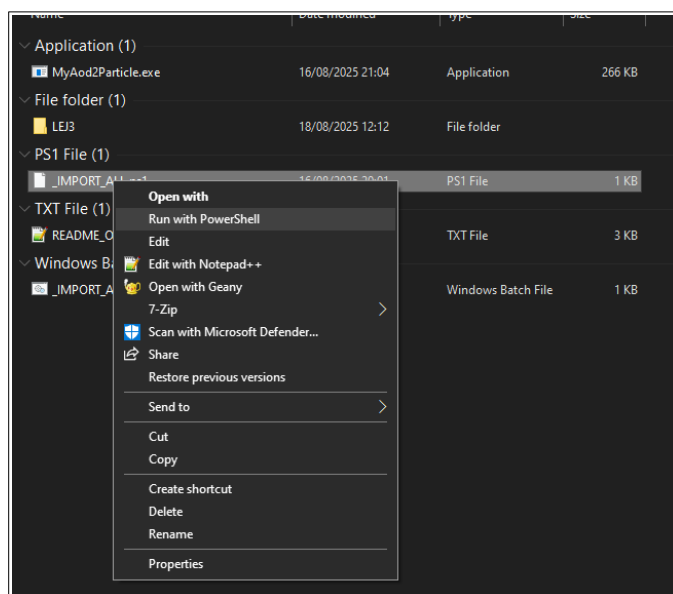


Figure 13: Możemy włączyć program przez dwuklik i wpisać nazwę katalogu do skompilowania, albo możemy włączyć skrypt `_IMPORT_ALL.ps1` przez powershella w celu skompilowania wszystkich katalogów z danymi particlesów.

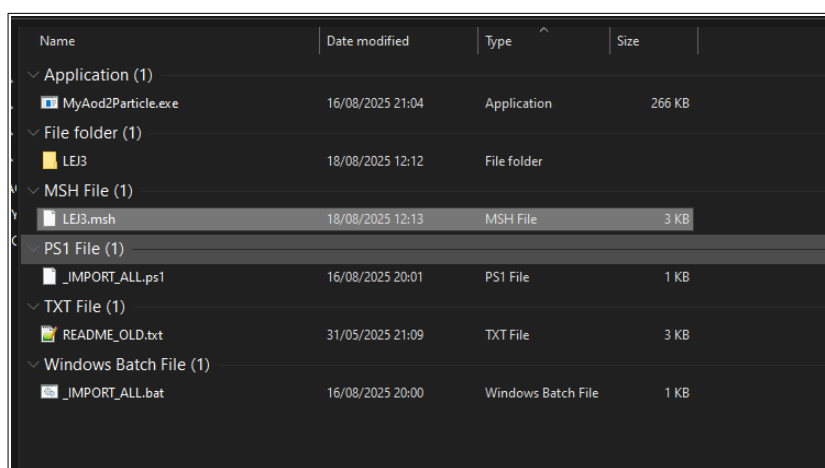


Figure 14: Gotowe - Particles jest gotowy do wrzucenia do gry.

c) Chcę przekonwertować ParticleEmitter z KnightShift do Earth 2160:

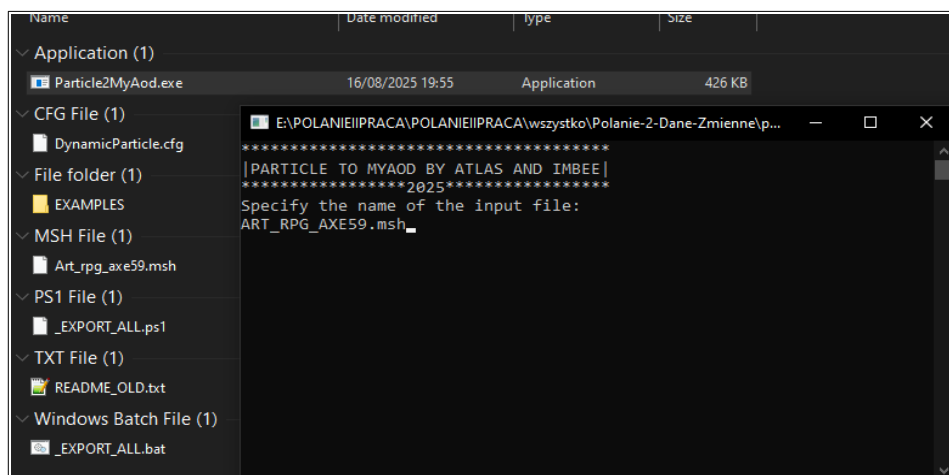


Figure 15: Włączam dwuklikiem Particle2MyAod.exe i wpisuję nazwę pliku .msh.

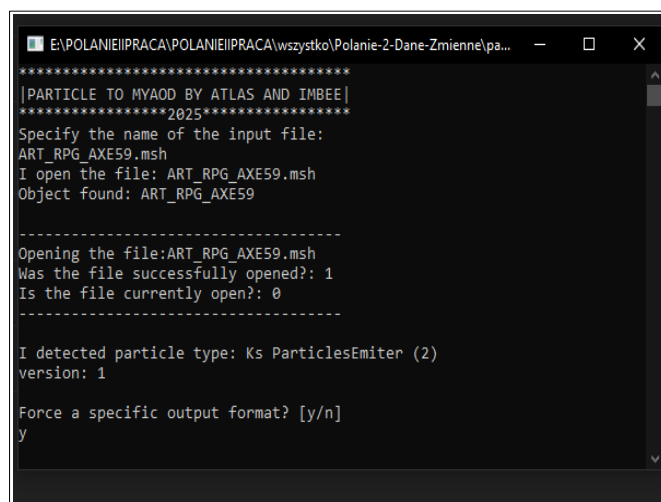


Figure 16: Wprowadzam y.

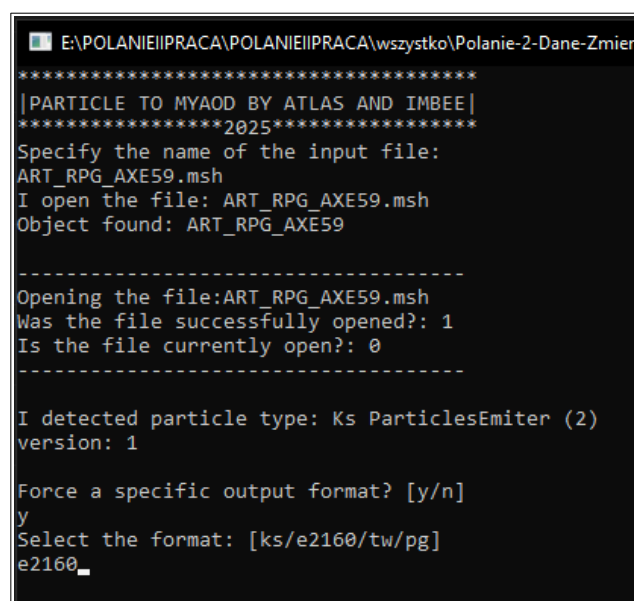


Figure 17: Wymuszam format e2160.

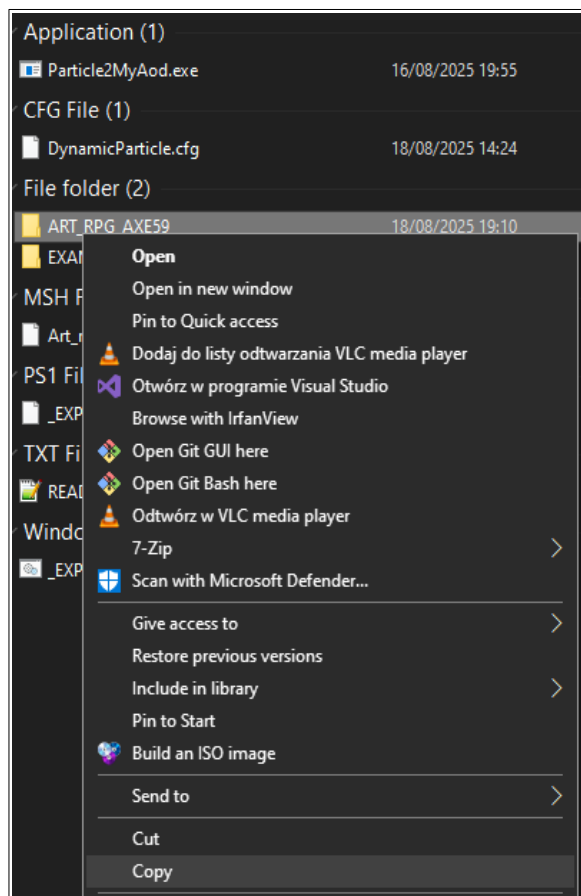


Figure 18: Kopiuję katalog z danymi particlesa do katalogu z MyAod2Particle.

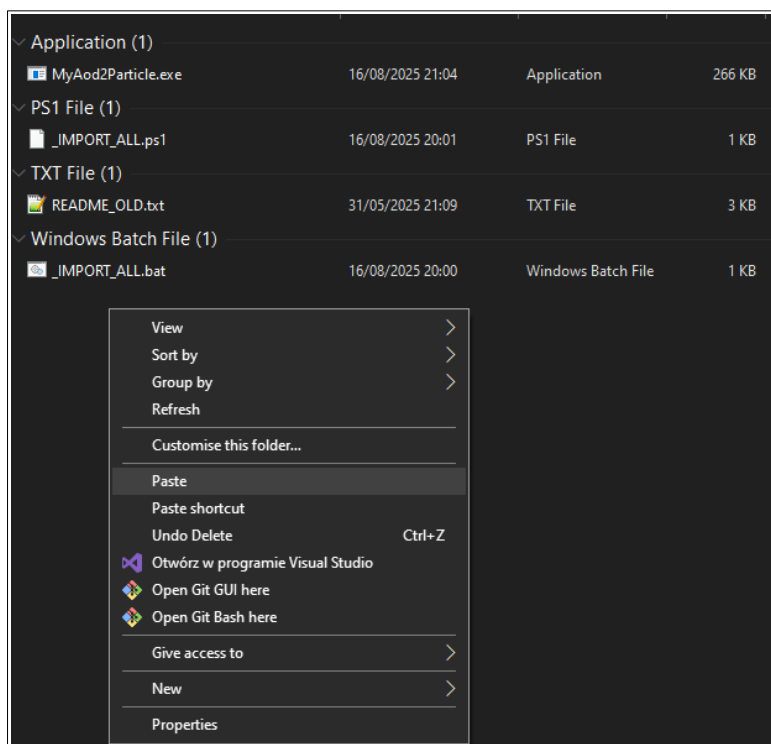


Figure 19: Wklejam.

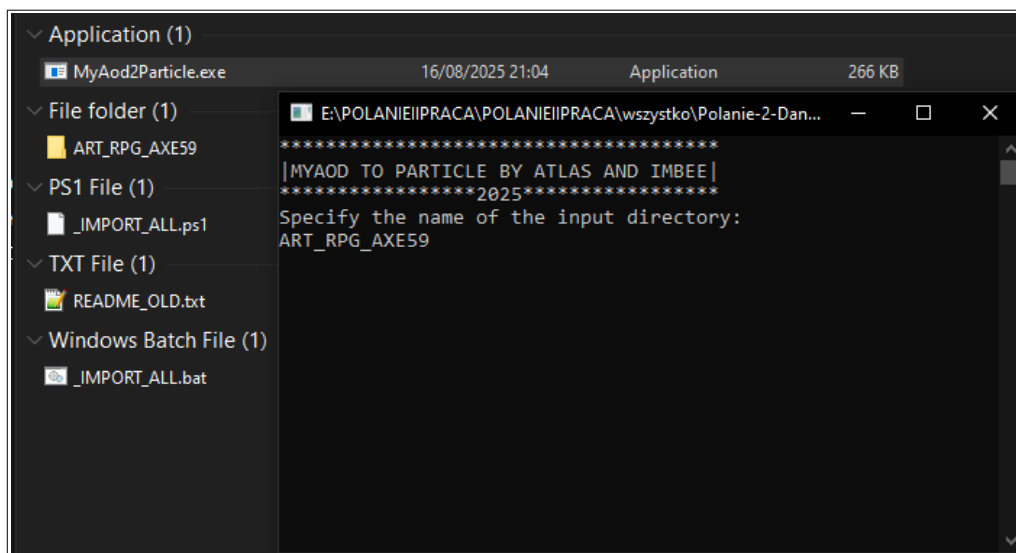


Figure 20: Włączam dwuklikiem program i wpisuję nazwę katalogu z danymi particlesa.

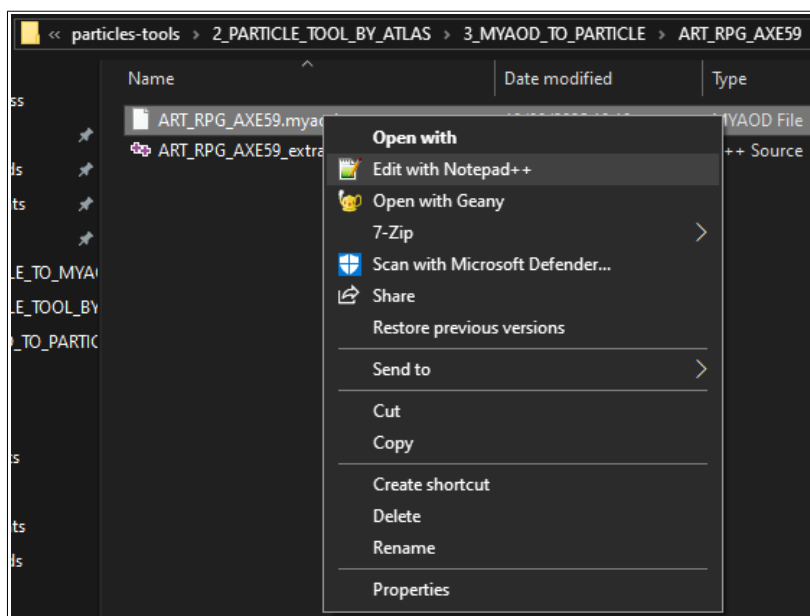


Figure 21: Wchodzę do katalogu z wypakowanymi danymi particlesa i sprawdzam zawartość pliku .myaod.

```

3418 {
3419     Emitter E4
3420     EmitterIndex 3
3421     Particle P4
3422     ParticleIndex 3
3423     annotation
3424     time 0.0000000000000000
3425     loopedEmission 2, 1, 0, 0.0000000000000000
3426     drawParticleEmitter 1
3427     simpleOneParticleSwitch 0
3428     2Dmask 0
3429     hardwareCursor 0
3430     stopInPartialPause 0
3431     finishMissile 0
3432     emitsGroupsSwitch 0
3433     emitsGroups
3434     onlyEmittedByOtherEmitterSwitch 0
3435 }
3436 gameRate 20
3437 endValue_0 0.0000000000000000
3438 endValue_1 0.0000000000000000

```

Figure 22: Sprawdzam i zapamiętuję gamerate.

Name	Date modified	Type	Size
ParticleEdit_NO_SSE.exe	17/08/2025 09:55	Shortcut	3 KB
ParticleEdit_SSE.exe	17/08/2025 09:55	Shortcut	3 KB

Figure 23: Włączam ParticleEdit.exe z E2160.

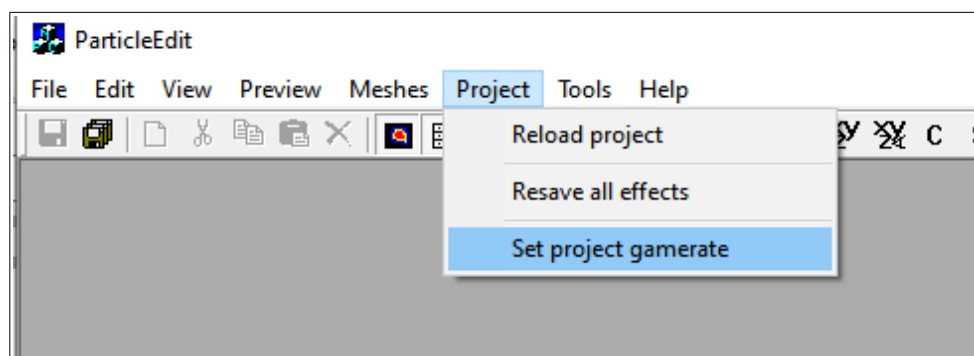


Figure 24: Namierzam i klikam opcję **Set project gamerate**.

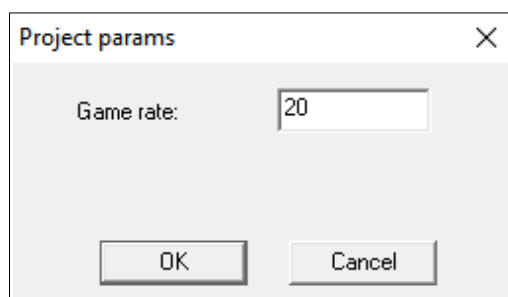


Figure 25: Ustawiam wartość na taką, którą wcześniej odczytaliśmy z pliku.

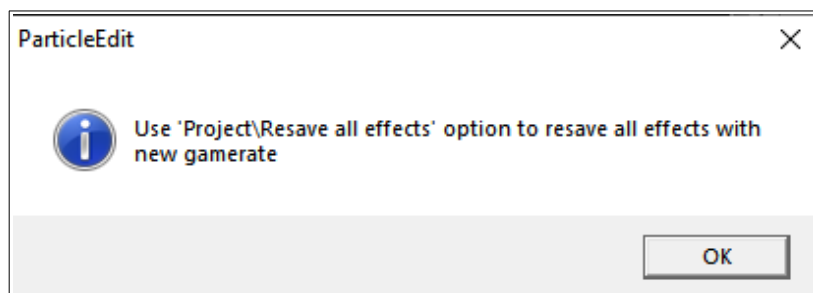


Figure 26: Wyświetla się taki komunikat.

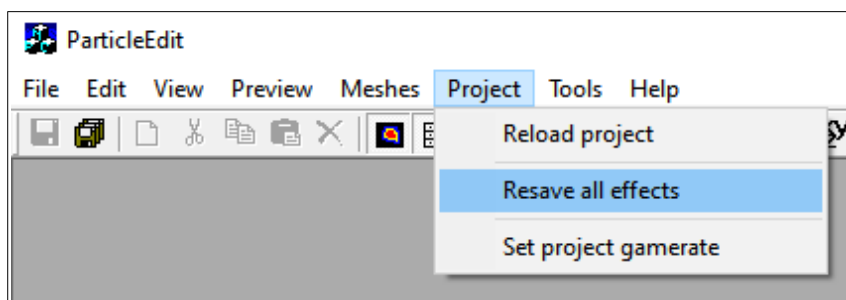


Figure 27: Namierzamy i klikamy **Resave all effects**. Aby operacja wykonała się poprawnie, należy chwilę poczekać.



Figure 28: Jak operacja się zakończy to wyłączamy program.

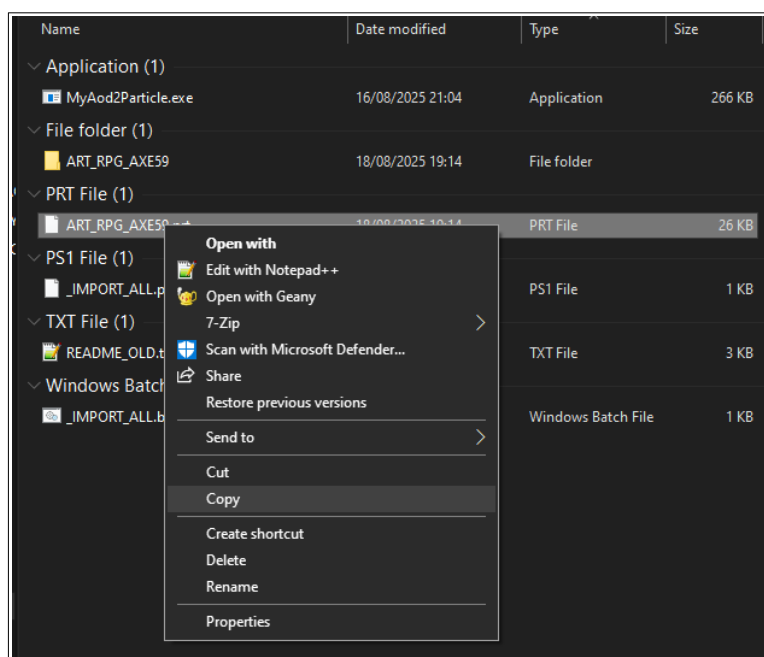


Figure 29: Kopiuje wyjściowy plik .prt.

Name	Date modified	Type	Size
Backups	08/07/2025 21:29	File folder	
Resources	10/06/2025 17:37	File folder	
1970.prt	18/08/2025 10:21	PRT File	19 KB
AMOEBE.prt	18/08/2025 10:27	PRT File	48 KB
ART_AMMO.prt	06/07/2025 12:21	PRT File	22 KB
ART_MINE.prt	07/07/2025 16:34	PRT File	18 KB
BLACK_HOLE_VORTEX.prt	18/08/2025 10:25	PRT File	20 KB
BM_7_2.prt	16/07/2025 22:19	PRT File	113 KB
Cursor01.prt	28/07/2025 14:35	PRT File	6 KB
e2160_prt1.prt	09/07/2025 20:30	PRT File	35 KB
e2160_prt2.prt	04/07/2025 12:27	PRT File	24 KB
Fire5.prt	18/08/2025 10:33	PRT File	34 KB
Geyser1.prt	07/07/2025 18:09	PRT File	28 KB
KOLUMNA.prt	19/07/2025 21:07	PRT File	43 KB
L_MI_EM_01_1.prt	21/06/2025 19:25	PRT File	12 KB
linked_e21.prt	09/07/2025 12:12	PRT File	5 KB
linked_e22.prt	08/07/2025 19:51	PRT File	5 KB
M8_TEMPLE_FX.prt	19/07/2025 21:32	PRT File	42 KB
particleslib.lprt	28/07/2025 13:29	LPRT File	1 KB
prt1.prt	04/07/2025 15:16	PRT File	32 KB
SaveEffects.log	01/08/2025 19:37	LOG File	29 KB
SG_OUT_OK.prt	07/07/2025 19:16	PRT File	81 KB
TEST.prt	05/06/2025 20:35	PRT File	5 KB
TEST.prt.bak	05/06/2025 20:33	BAK File	5 KB
TEST_or.prt	05/06/2025 20:46	PRT File	10 KB
TTEE.prt	08/06/2025 12:55	PRT File	8 KB
TTEE1.prt	08/06/2025 15:16	PRT File	8 KB
TTEE1.prt.bak	08/06/2025 15:14	BAK File	8 KB

Figure 30: Namierzam katalog z plikami .prt programu ParticleEdit z Earth 2160.

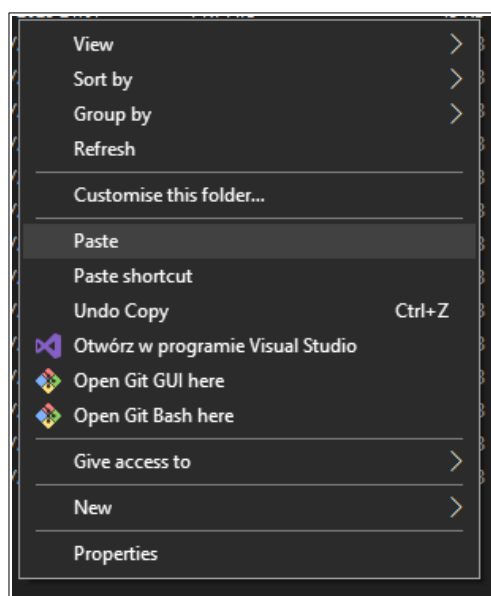


Figure 31: Wklejam.



	ParticleEdit_NO_SSE.exe	17/08/2025 09:55	Shortcut	3 KB
	ParticleEdit_SSE.exe	17/08/2025 09:55	Shortcut	3 KB

Figure 32: Ponownie uruchamiam ParticleEdit z Earth 2160.

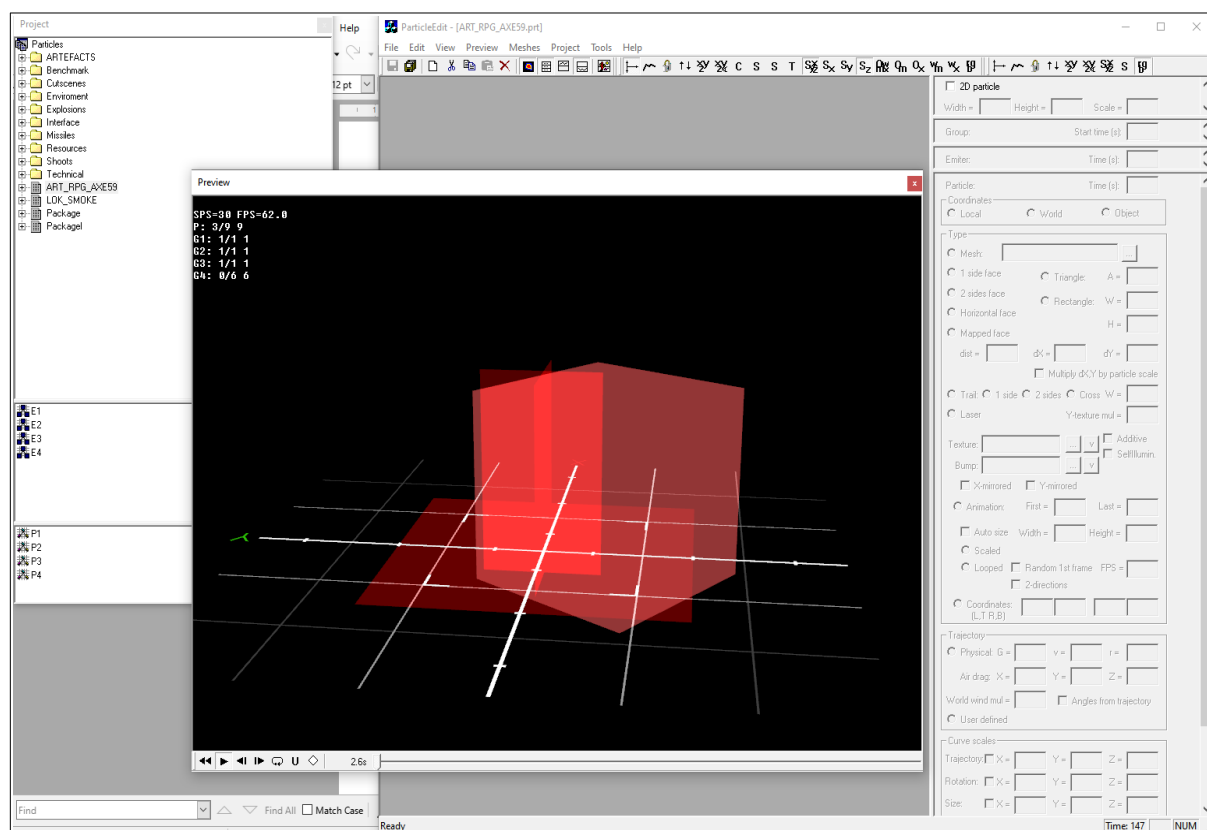


Figure 33: Jak widzimy particle się uruchomił ale, żeby pozbyć się tych czerwonych tekstur należy przekonwertować .tex na .tga i wrzucić tekstury do katalogu Textures w katalogu z grą E2160.

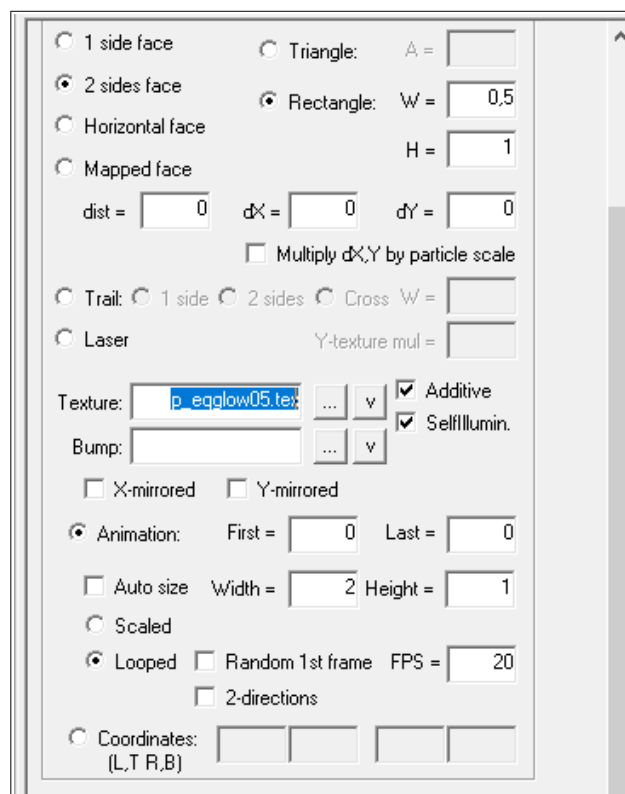


Figure 34: Można przeskakiwać po obiektach ParticleEmitera i zmieniać nazwy tekstur oraz meshy.

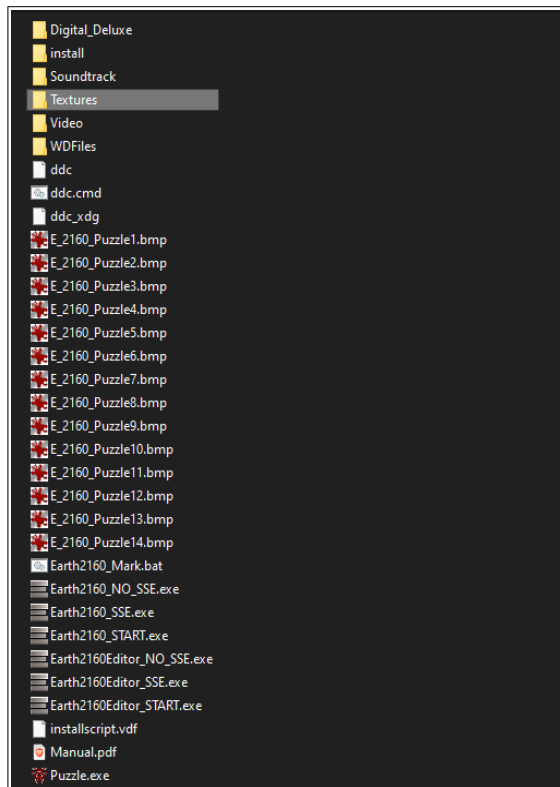


Figure 35: Katalog Textures tworzymy w folderze z grą Earth 2160 i wrzucamy tam nasze texture w formacie tga. W ten sposób nie trzeba pakować tych plików do WD.

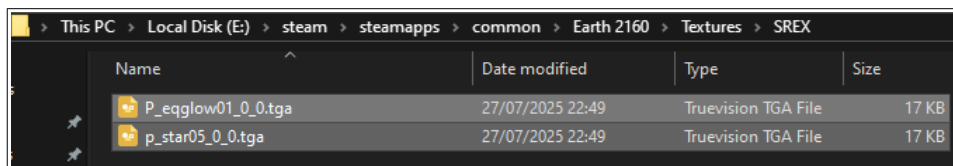


Figure 36: Ja sobie wrzuciłem takie texture.

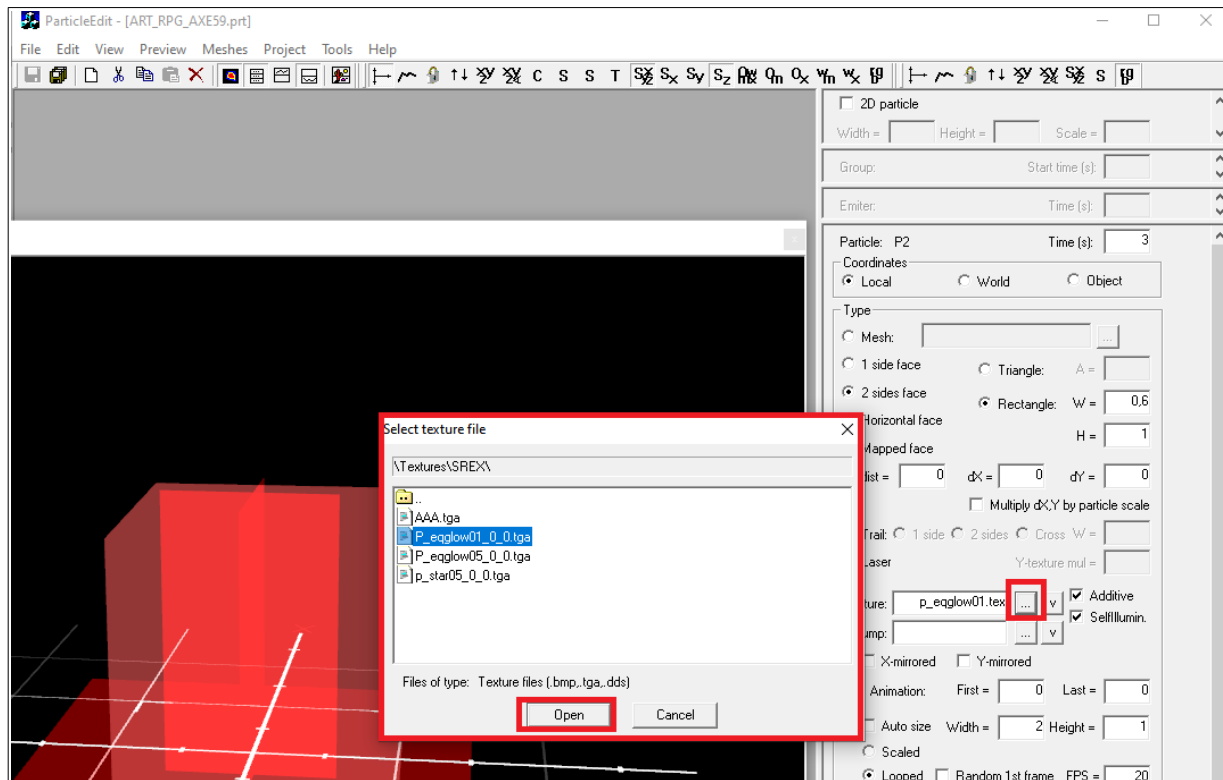


Figure 37: Wybieramy texture tga z katalogu z teksturami. Robimy tak dla każdej texture tex. Przy okazji możemy zmienić też mesha na np. Sphere.

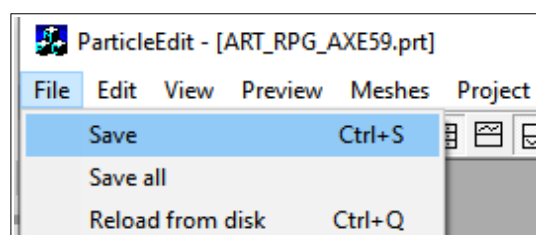


Figure 38: Zapisujemy particlesa.

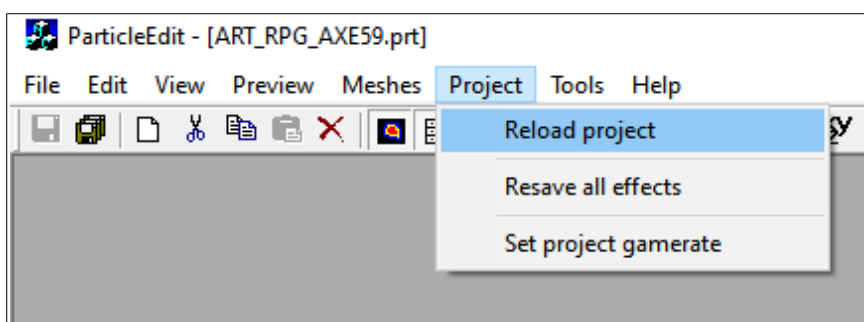


Figure 39: Aby odświeżyć projekt w celu zobaczenia zmian, możemy kliknąć opcję **Reload project**.

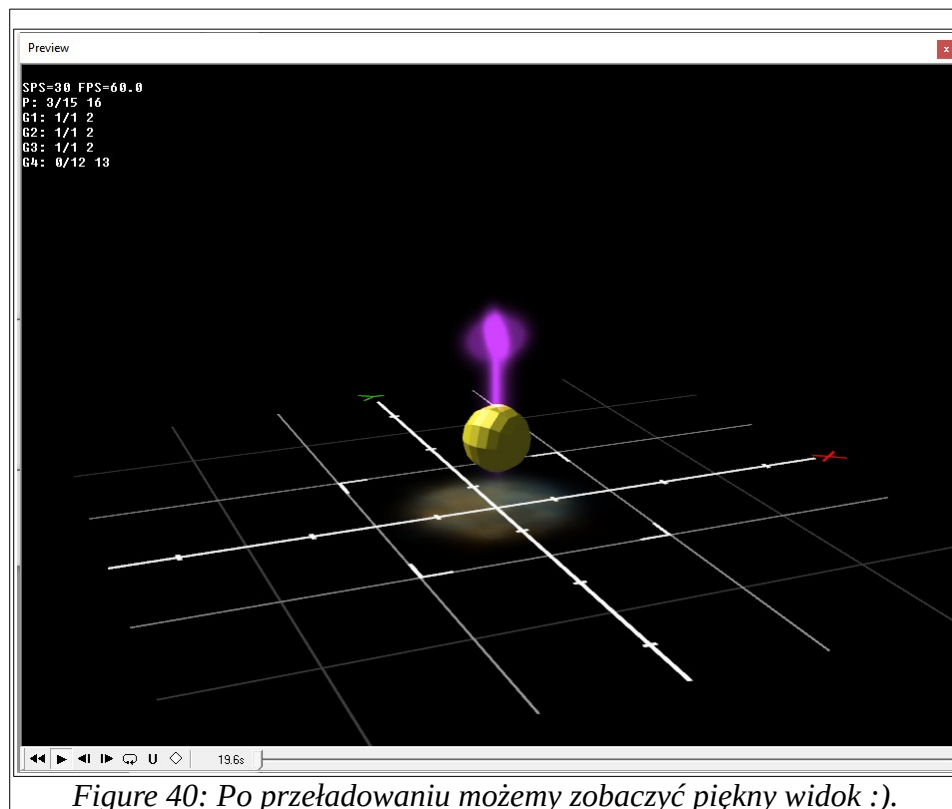


Figure 40: Po przeładowaniu możemy zobaczyć piękny widok :).

d) Chcę dokonać zmian w ParticleEmitterze z KnightShift.
Eksportujemy plik do formatu E2160 jak w punkcie c). Edytujemy według uznania.

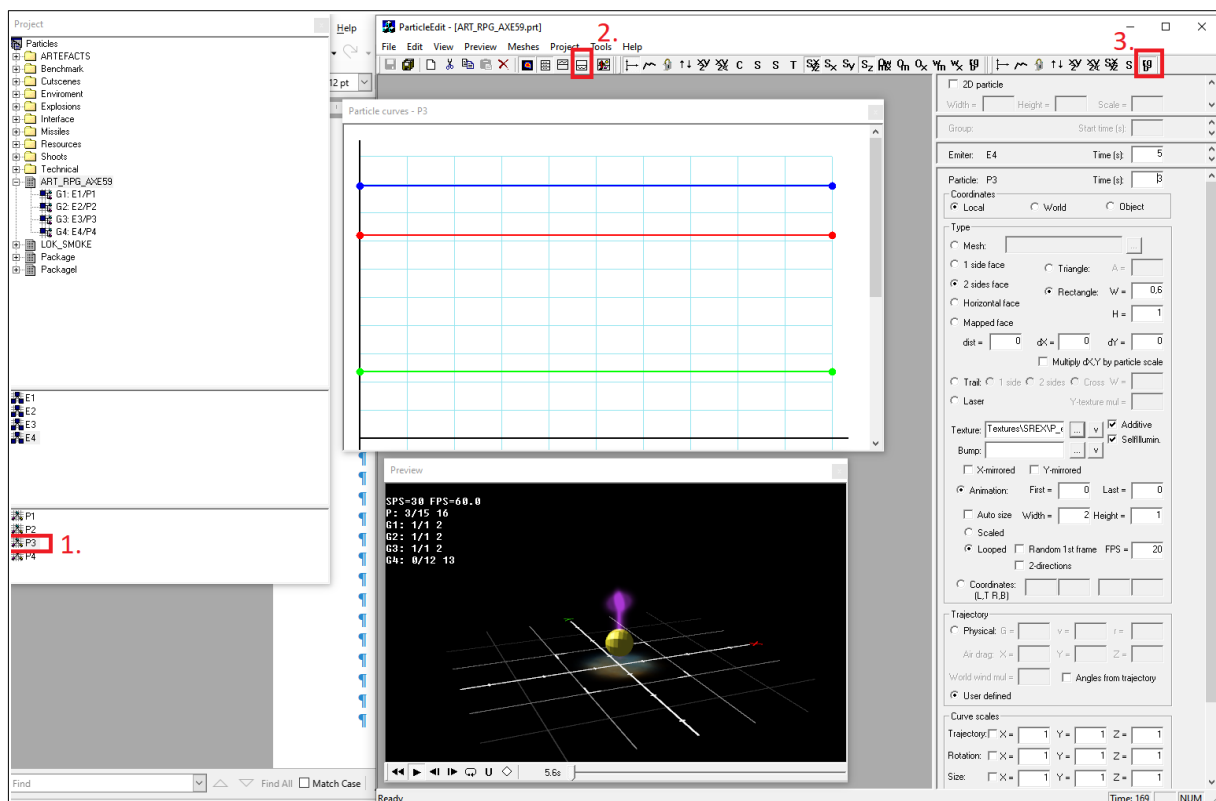


Figure 41: Teraz np. namierzamy to co chcemy zmienić i dokonujemy zmian. Ja przykładowo zmienię kolor tego obramowania broni.

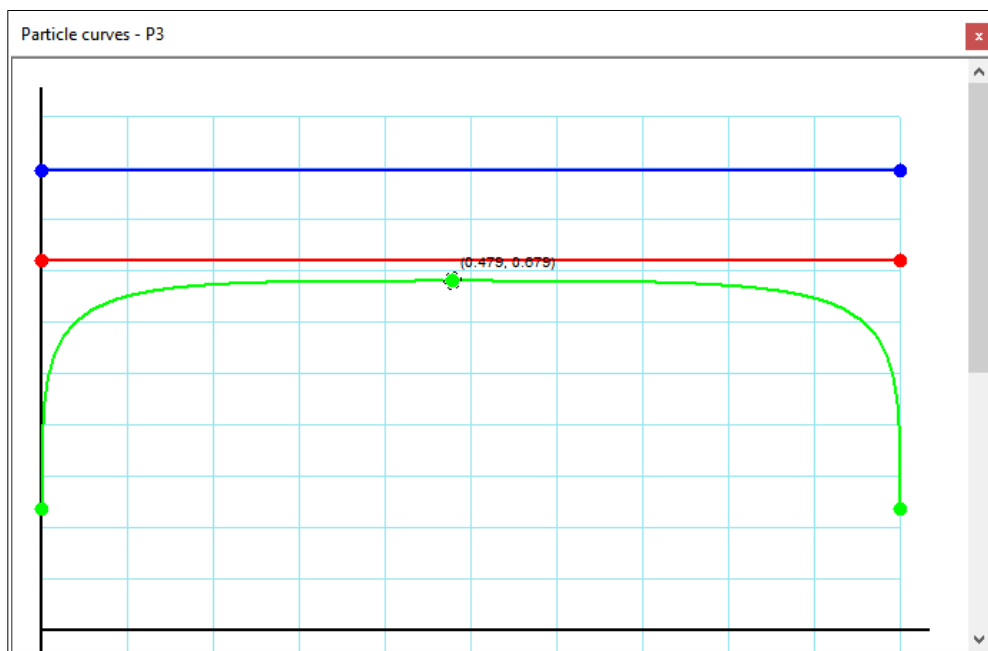


Figure 42: Manipuluję punktami.

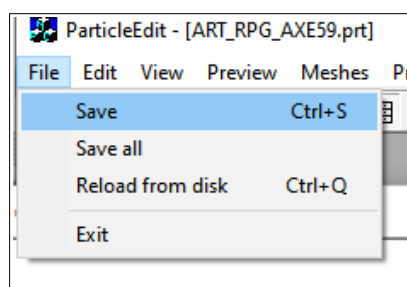


Figure 43: Zapisuję.

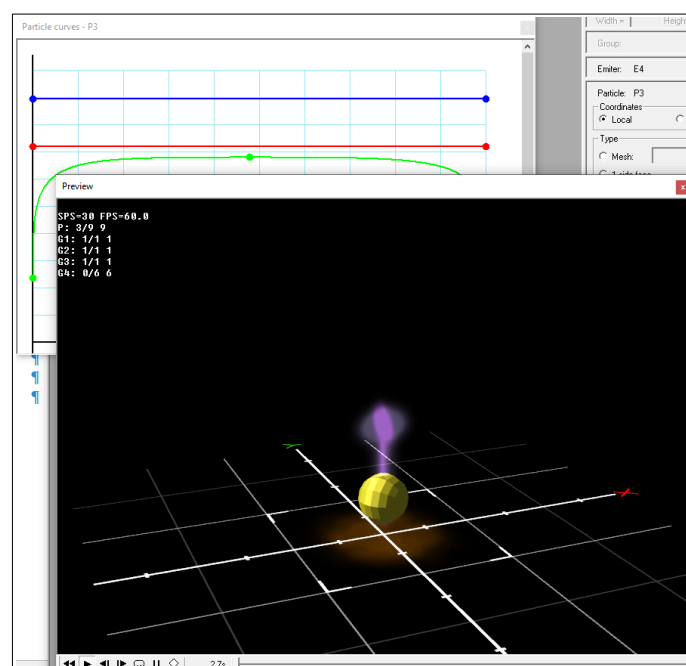


Figure 44: Obrazowanie się zmieniło na kolor biały.

Jeżeli chcemy z powrotem wrzucić ten plik do KnightShift to musimy przywrócić wpisy o poprzednich teksturach .tex i meshach. Można to zrobić ParticleEditem w tym momencie albo podczas powrotnej konwersji do formatu .myaod przez edytor tekstu np. Notepad++.

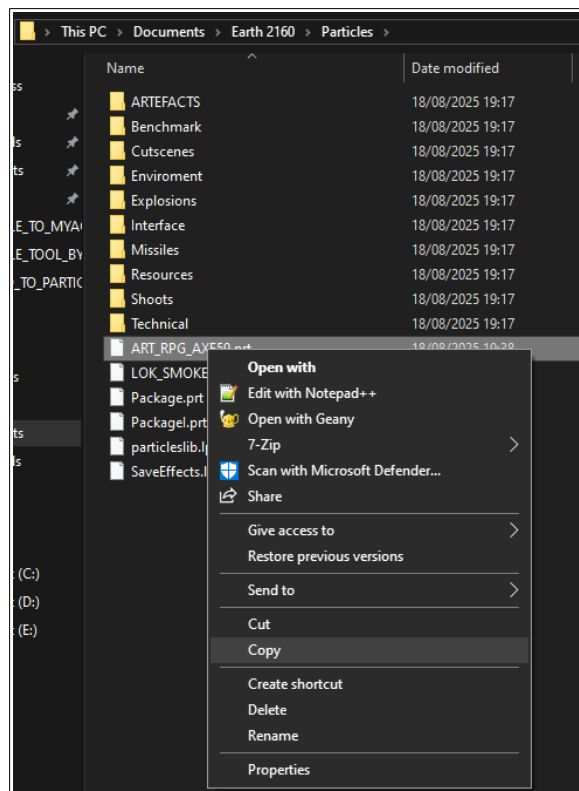


Figure 45: Kopiujemy plik .prt z katalogu particlesów programu ParticleEdit.

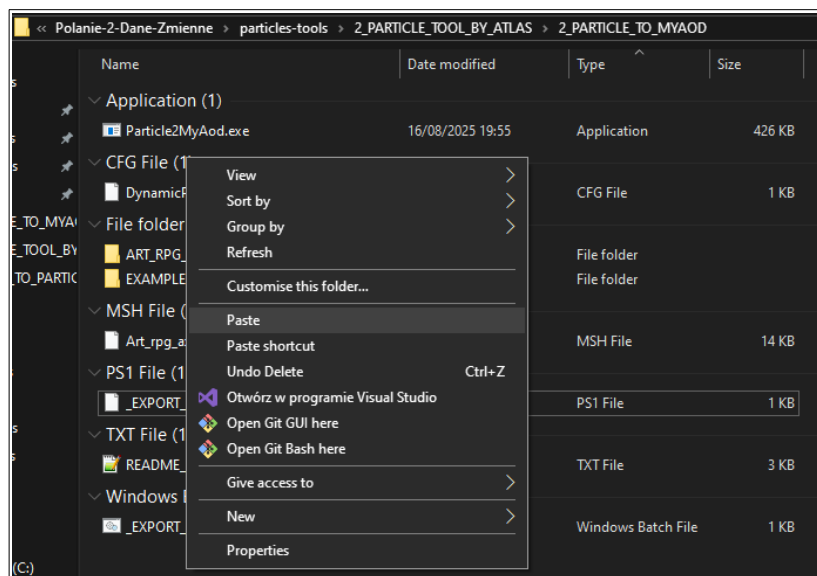


Figure 46: Wklejamy do katalogu Particle2MyAod.

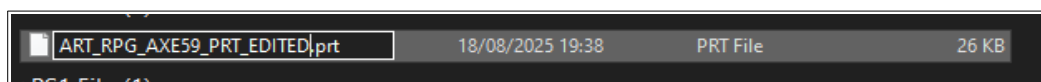


Figure 47: Zmieniamy profilaktycznie nazwę pliku, żeby katalogi wyjściowe się nie pomieszały.

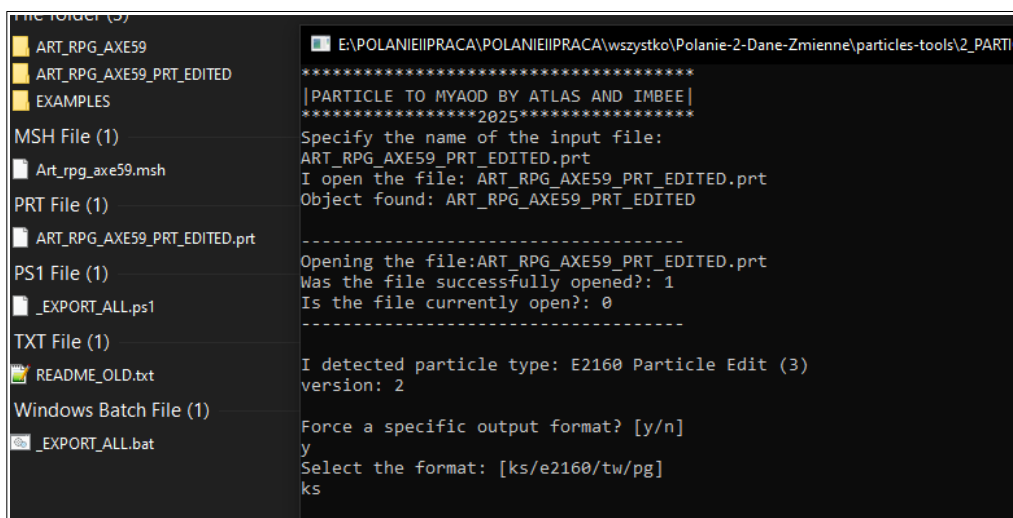


Figure 48: Eksportujemy wymuszając format ks.

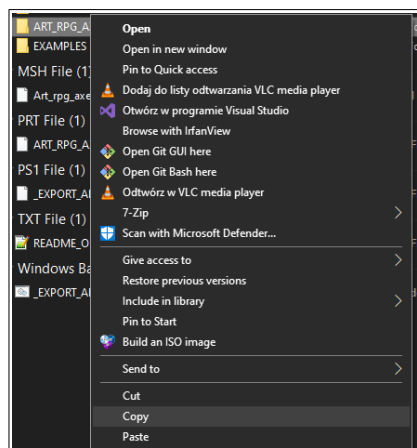


Figure 49: Kopiujemy katalog, który powstał.

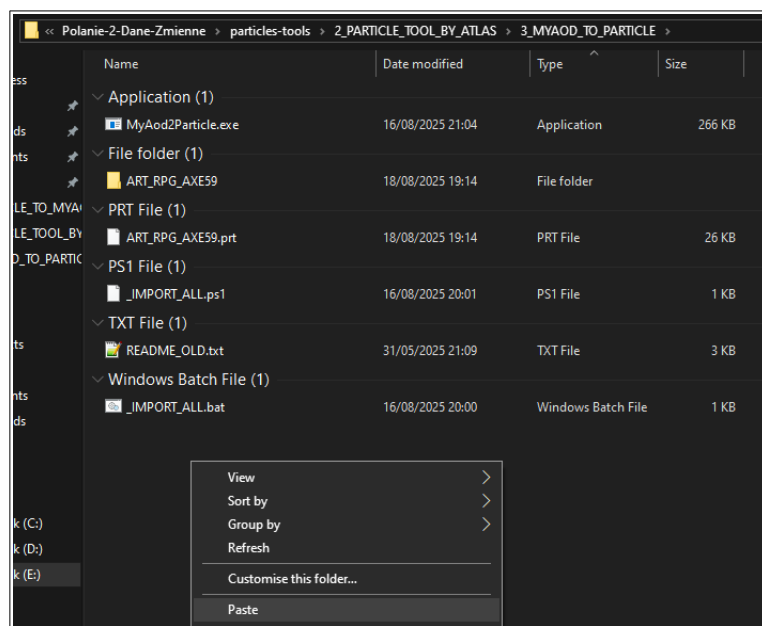


Figure 50: Wklejamy go do katalogu MyAod2Particle.

```

E:\POLANIEIIPRACA\POLANIEIIPRACA\wszystko\Polanie-2-D
*****
|MYAOD TO PARTICLE BY ATLAS AND IMBEE|
*****2025*****
Specify the name of the input directory:
ART_RPG_AXE59_PRT_EDITED_

```

Figure 51: Importujemy.

MSH File (1)			
ART_RPG_AXE59_PRT_EDITED.msh	18/08/2025 19:53	MSH File	14 KB
PRT File (1)			

Figure 52: Gotowe - można wrzucić plik do KnightShift.

e) Chcę uruchomić plik .prt z 3D ParticleGen Visual FX (steam) w ParticleGenie z Two Worlds:

Name	Date modified	type	Size
Application (1)			
Particle2MyAod.exe	16/09/2025 10:55	Application	176 KB
CFG File (1)			
DynamicParticle.cfg			
File folder (2)			
1970			
EXAMPLES			
PRT File (1)			
1970.prt			
PS1 File (1)			
_EXPORT_ALL.ps1			
TXT File (1)			
README_OLD.txt			
Windows Batch File (1)			
_EXPORT_ALL.bat			

Figure 53: Włączam program **Particle2MyAod.exe**.
Wprowadzam **nazwę pliku**, potem wprowadzam **y** wymuszając
inny format i wprowadzam **tw**.

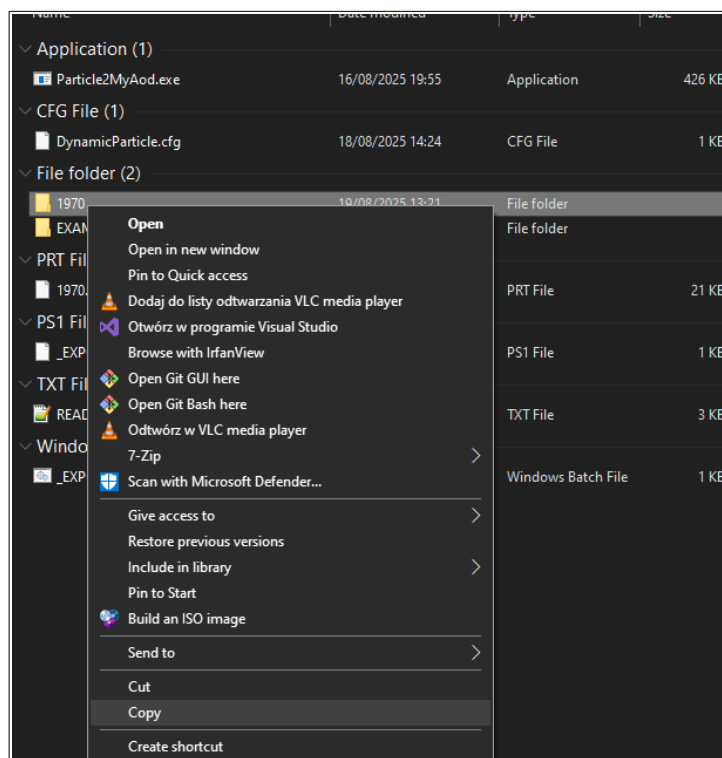


Figure 54: Kopiuję uzyskany katalog.

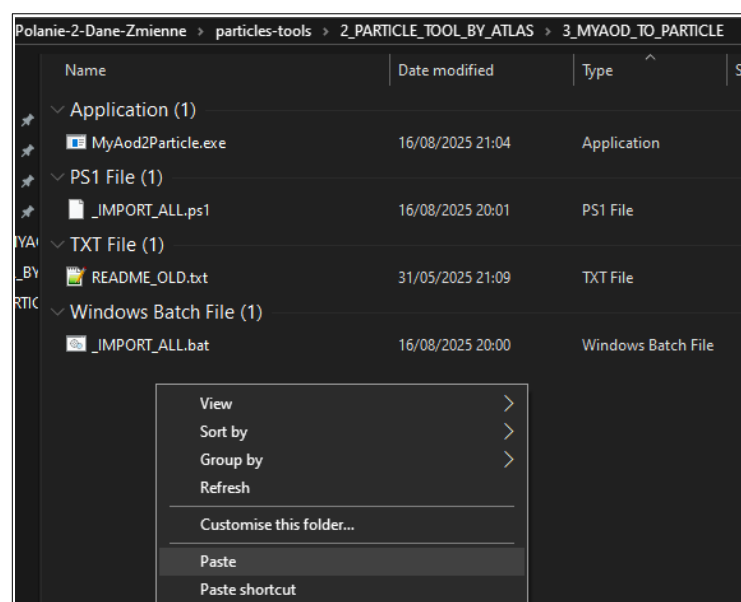


Figure 55: Wklejam do katalogu z **MyAod2Particle.exe**.

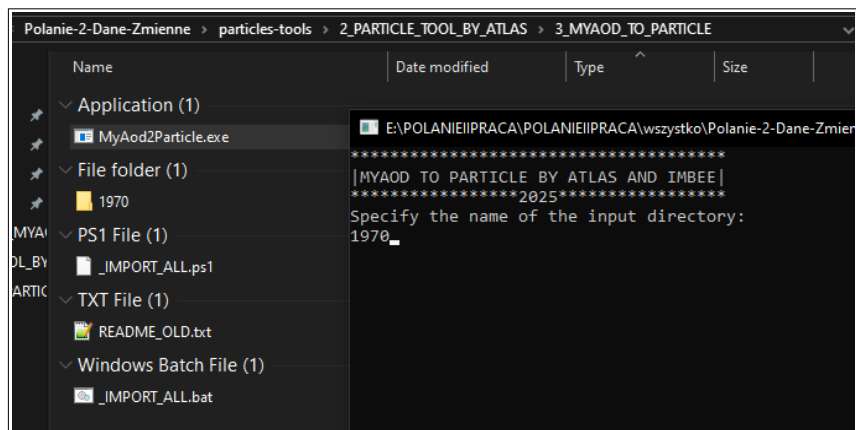


Figure 56: Włączam **MyAod2Particle** i wprowadzam do niego nazwę wklejonego katalogu.

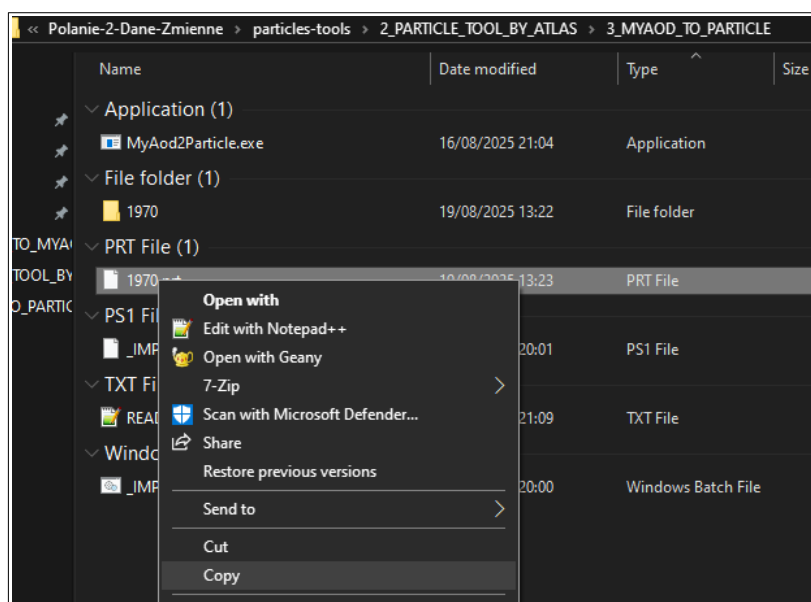


Figure 57: Kopiaję uzyskany plik **.prt**.

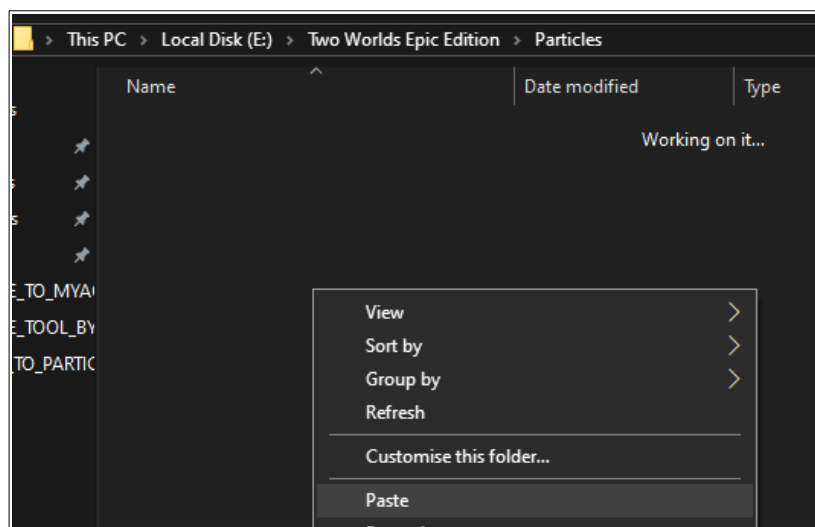


Figure 58: Wklejam do katalogu **Particles** w folderze z grą **Two Worlds**.

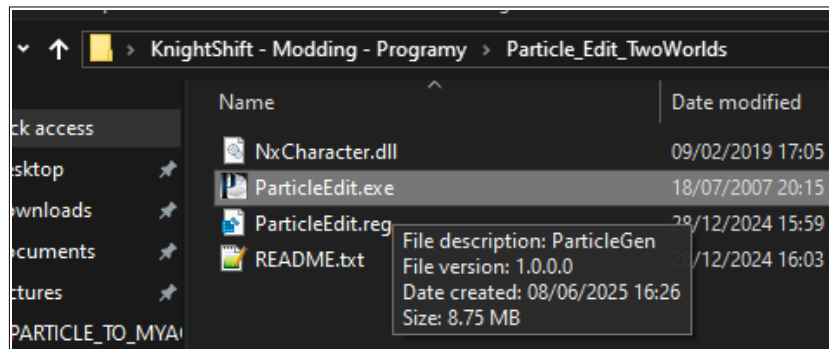


Figure 59: Uruchamiam **ParticleGena** z **Two Worlds SDK**.

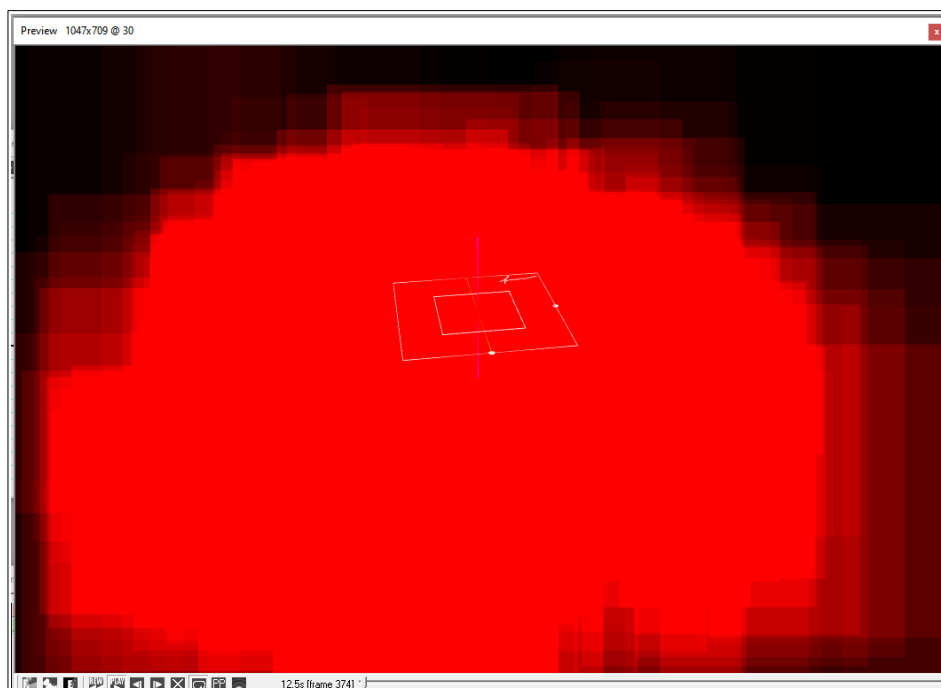


Figure 60: Jak widzimy po uruchomieniu particlesa w programie - plik działa jednak trzeba jeszcze przerzucić odpowiednią teksturę z 3D Particle Gena do Two Worlds.

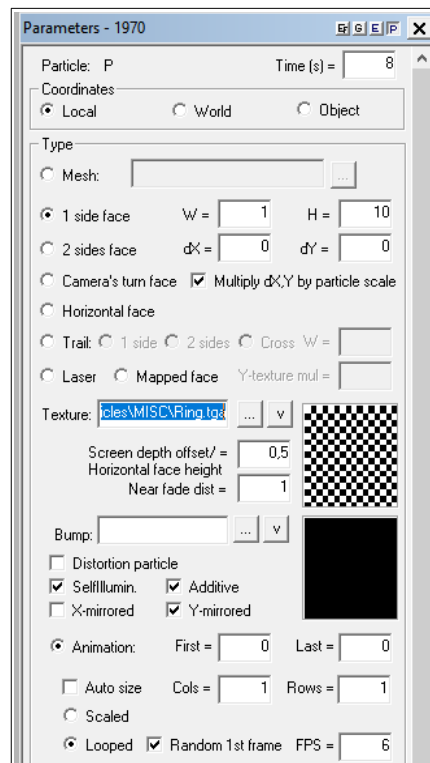


Figure 61: Możemy namierzyć uruchamianą teksturę w parametrach obiektów.

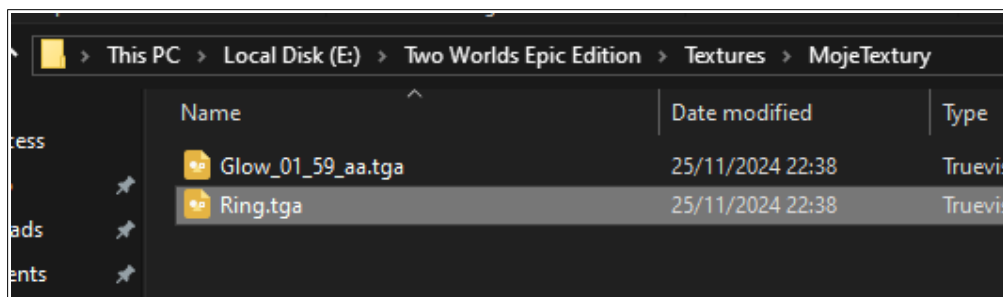


Figure 62: Wrzucamy do katalogu Textures w Two Worlds wymaganą teksturę (ja sobie zrobiłem dodatkowy katalog wewnątrz o nazwie MojeTexturey).

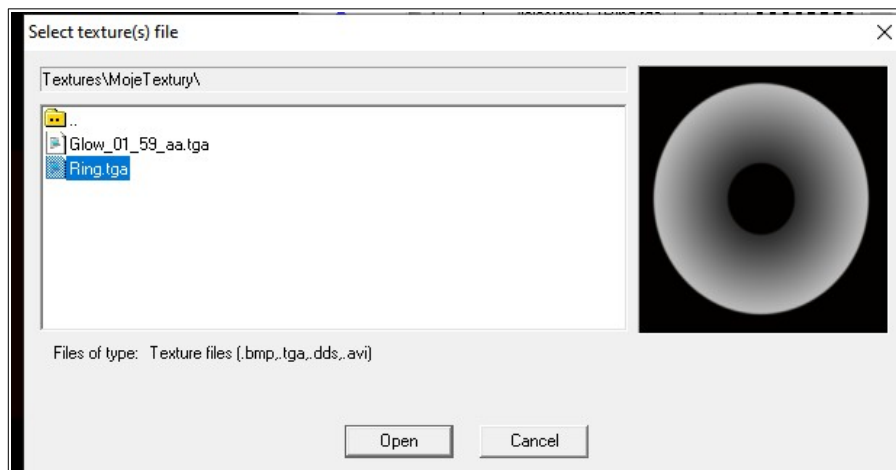


Figure 63: W programie do particlesów wybieramy dorzuconą teksturę.

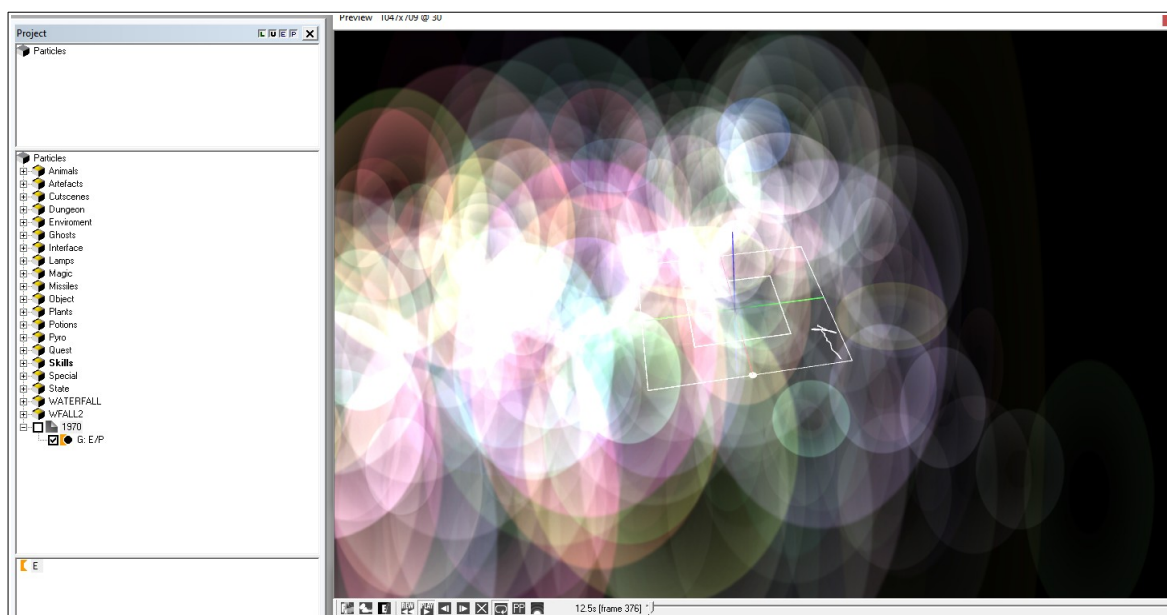


Figure 64: Po zapisaniu particlesa i zrestartowaniu programu możemy zauważyć efekt końcowy.