

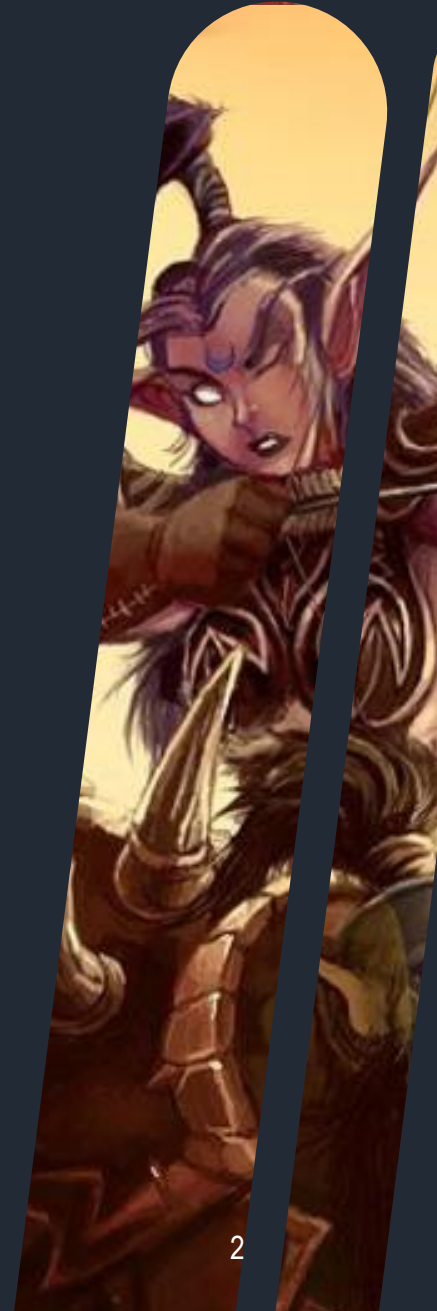


Reverse-Engineering

Extracting Game Assets from Warcraft 1 (MacOS)

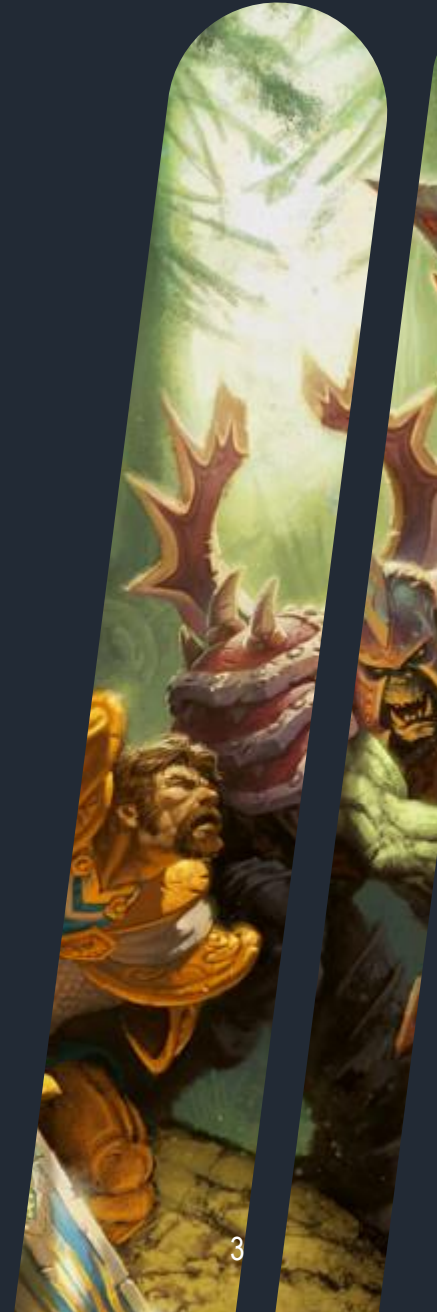
Overview

- Where we left off
- What we have done since then
- What we would want to try if we had the time




Where we left off

- rewritten some extraction tools in Python
- extracted text & audio
- developed some ideas for helpful tools



What we had as a starting point



Warcraft: Orcs & Humans

[Files/Versions](#) [Screenshots](#)

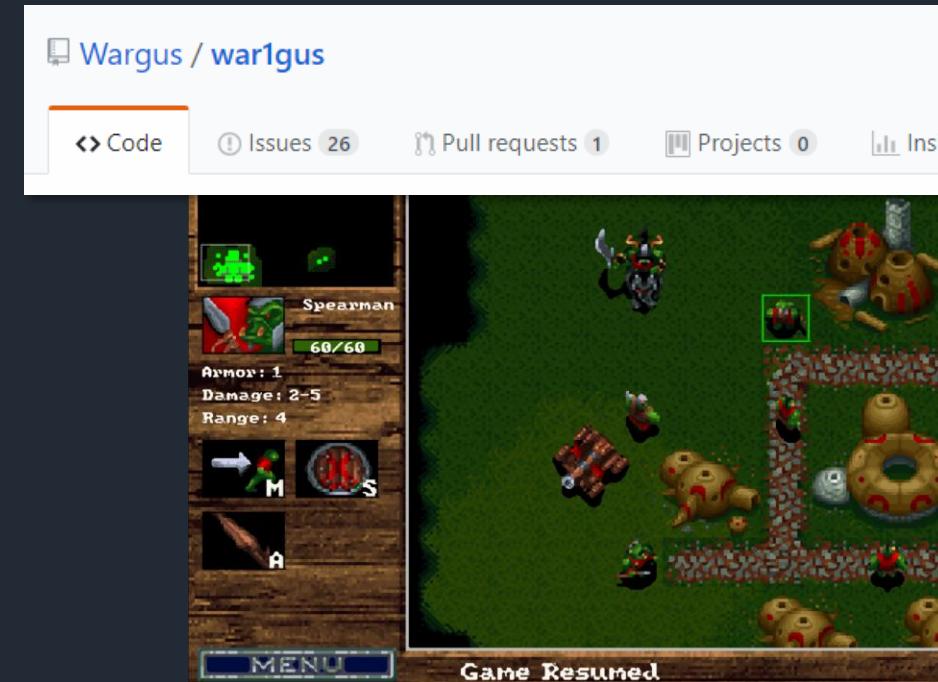
Version History

Ver.	Date	Comment/Download	List File	Archive Version
Warcraft 1 PC				
	Oct 04, 1994	Warcraft Demo 1	Demo	Demo
	Oct 06, 1994	Warcraft Demo 2	Demo	Demo
	Oct 06, 1994	Warcraft Interplay Demo	Demo	Demo
1.12	Nov 03, 1994	Retail Floppy	Retail	Retail
1.12	Nov 03, 1994	Retail CD-ROM	Retail	Retail
1.12	Nov 23, 1994	Warcraft Demo 3	Retail	Demo
1.13				
1.14				
1.15	Dec 27?, 1994	Warcraft Patch 1.15		
1.16	Jan 25, 1995	Warcraft Patch 1.16		
1.20	Mar 06, 1995	Warcraft Patch 1.20		
1.21	Mar 15, 1995	Creative Labs Demo CD 1.21		

Extras

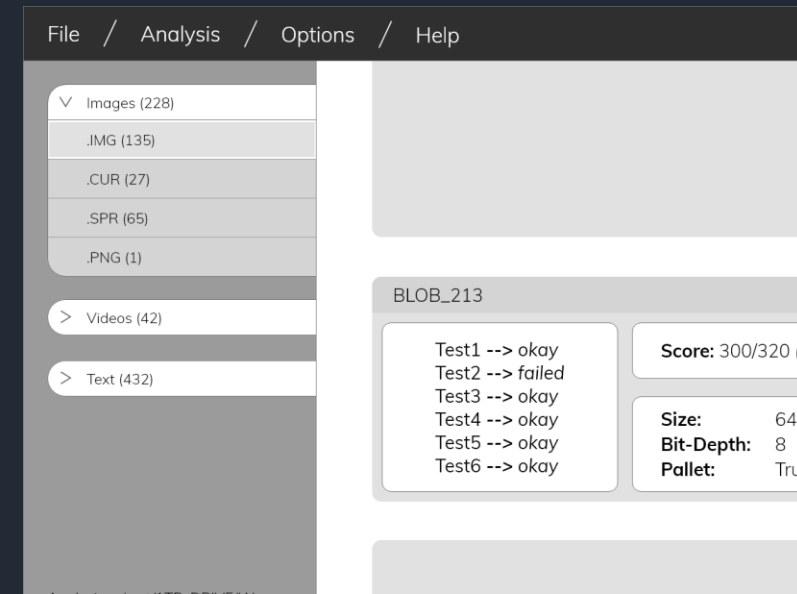
- [How Warcraft 1 files work \(PDF\)](#)
- [Warcraft 1 Insider's Guide \(1995-08-27\)](#)
- [All DATA.WAR files \(except MPlayer and TEN\)](#)

BIN-FILE



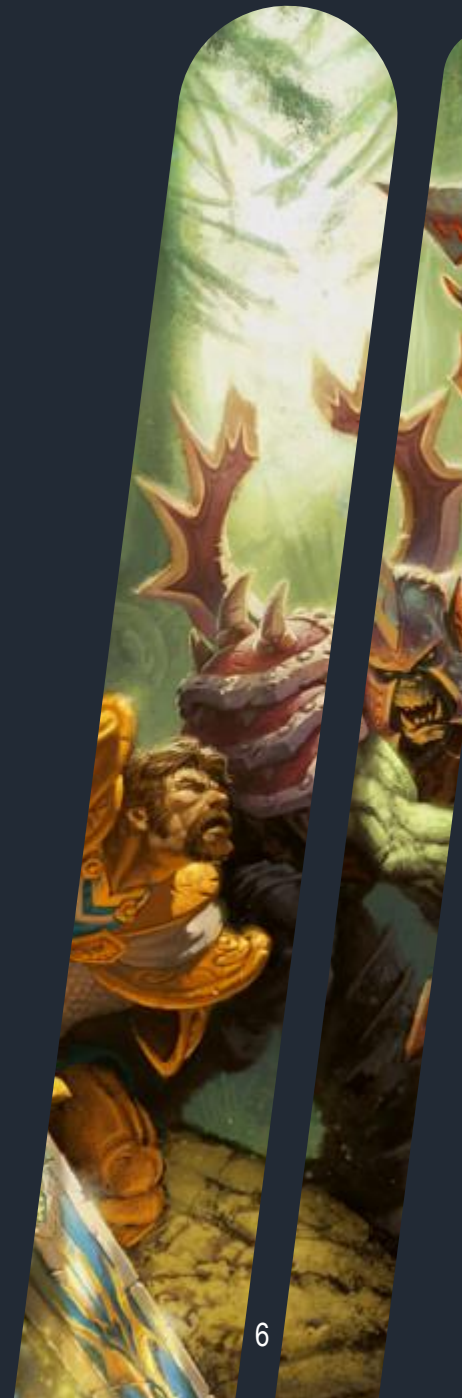
What we wanted to do since we left off

- develop the presented tools (for better ease-of-use)
 - evaluate developed indicators (e.g. For JPEGs)
- extract assets from the Binary (as it is the original task)



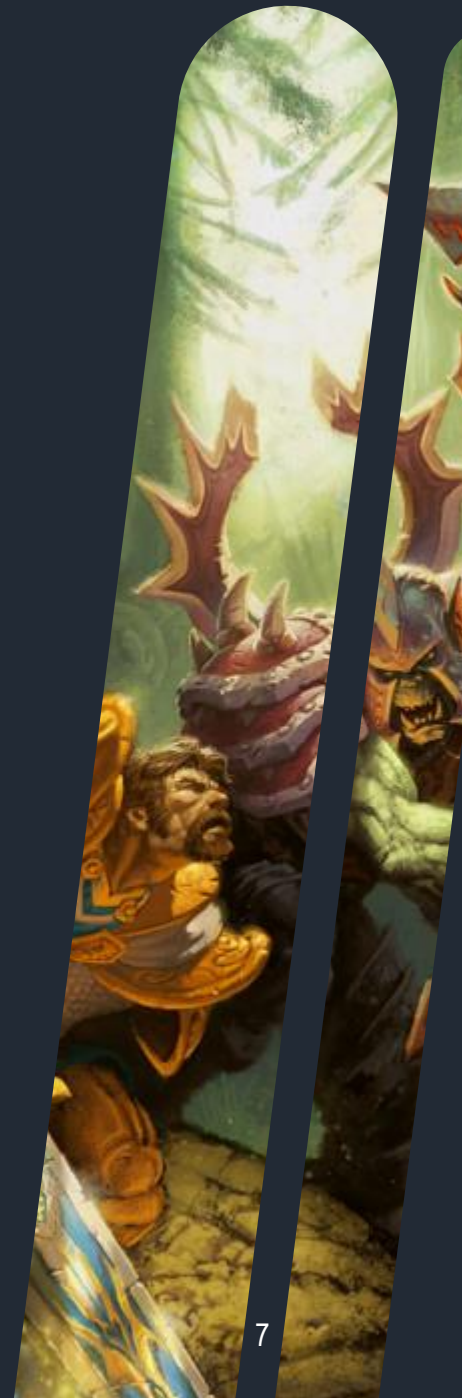
What we actually did

- extracted sprites and image assets
- extracted image paletts
- endianness is now taken into account
- developed a frontend for our tool(s)



What we could not do

- developed more & better metrics
 - time constraints
- idea: let the computer discover the metric
 - neural networks as qualifier
 - LSTM – networks for analyzing continuous data
- image metrics
 - high repetition data
 - reason about size-bytes

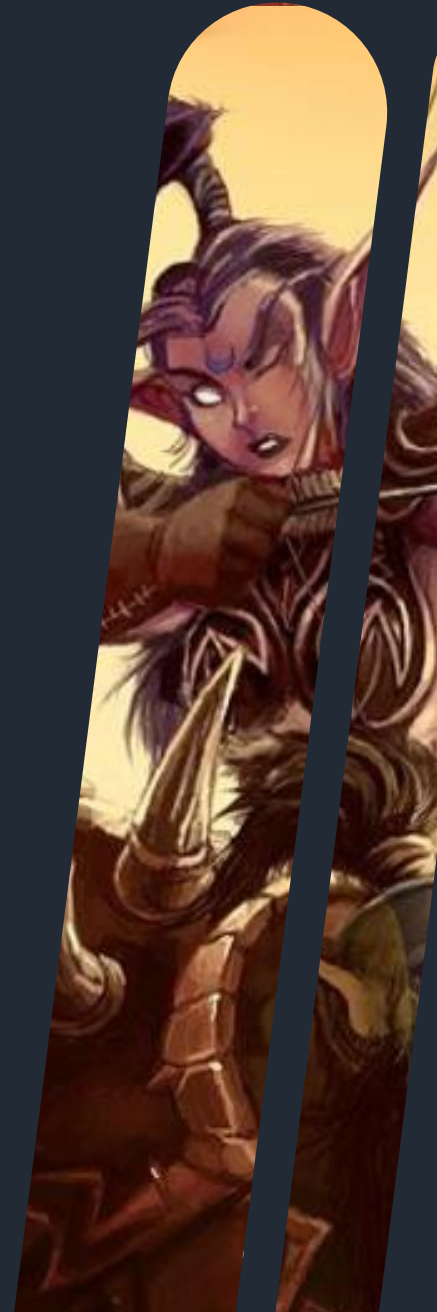


Demo Time

-- frontend --

29.01.2019

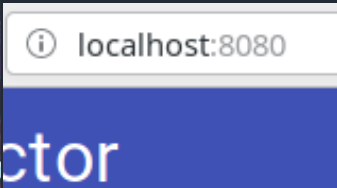
Noel Danz & Raoul Baron



Demo Time

-- backup --

```
7/mnt/1TB_DRIVE/OneDrive/ONI/rev
> npm run dev
> reverseengineering@0.1.0 dev
> cross-env NODE_ENV='development'
```



WarExtractor

UPLOAD WARFILE

File Table

Name	Offset	FileSize	Compressed	Prediction
------	--------	----------	------------	------------

Rows per page: 10 0-0 of 0



```
Idle predicting 110
Idle predicting 111
Idle predicting 112
Idle predicting 113
Idle predicting 114
Idle predicting 115
Idle predicting 116
Idle predicting 117
Idle predicting 118
Idle predicting 119
Idle predicting 120
Idle predicting 121
Idle predicting 122
Idle predicting 123
Idle predicting 124
Idle predicting 125
```



File Table 583 files found			
Name	Offset	FileSize	Compressed
0	4672	0	⊗
1	4672	0	⊗
2	4672	0	⊗
3	4672	0	⊗
4	4672	0	⊗
5	4672	0	⊗
6	4672	325381	⊗
7	330053	510694	⊗
8	840747	393723	⊗
9	1234470	467406	⊗

File Table | 583 files found

Name	Offset	FileSize	Compressed	Prediction
0	4672	0	⊗	GUESS: PLACEHOLDER
1	4672	0	⊗	GUESS: PLACEHOLDER
2	4672	0	⊗	GUESS: PLACEHOLDER
3	4672	0	⊗	GUESS: PLACEHOLDER
4	4672	0	⊗	GUESS: PLACEHOLDER
5	4672	0	⊗	GUESS: PLACEHOLDER
6	4672	325381	⊗	GUESS: AUDIO
7	330053	510694	⊗	GUESS: AUDIO
8	840747	393723	⊗	GUESS: AUDIO
9	1234470	467406	⊗	GUESS: AUDIO

Rows per page: 10 1-10 of 583

16373407 17590 ⊗ GUESS: AUDIO

Entropy of blob: 0.869

Choose file type

AUDIO

SAVE CHOICE

Blob: 475

Choose Test

Check if text

Check if audio

Check if image

Check if sprite

Audio player

Play

EDIT

TEST

Blob: 475

Entropy of blob: 0.8698215503159245 - Guessed file type is AUDIO

Choose file type

UNKNOWN

SAVE CHOICE

EDIT

TEST

If we had more time we would

- try to find more & better metrics
 - implement the neural network
- better UX for the tool
 - image alignment tool
 - full circle: hex editor
- make it more universal
- make it more robust

Thanks!

