Project SMD

4. scenario.chk

why do you ask
2013. 9. 10. 20:43

This document is a course-style log of developing the StarCraft editor .
Since I am also in the position of learning, what is the point of calling this a 'course';
This is a project to make a map editor, realizing that making a map editor involves under standing the overall structure of the game.

http://blog.naver.com/whyask37

---------- ----------

Main Sources:
http://www.staredit.net/starcraft/CHK
http://cafe.daum.net/rpgguild/6cWR/158
https://code.google.com/p/vgce/source/browse/trunk/docs/Blizzard/Starcraft/chkFormat.txt

Well anyway, I can't explain more than the data above, so learn about each paragraph in t he chk with the 3 texts above.

Anyway, if I write a course on the chk format, it will be more difficult than the above two. If you'r e a programmer, you'll understand all three of the above.

Clearly, the staredit material is better, but the easy-to-understand material is the rpgguild materi al.
I started with the rpgguild material, and I am referencing it in parallel with the SEN material.
vgce is an analysis of the chk format from the point of view of reversing. I don't know either.

See above that the chk file is like this
Here we will talk about how to read a chk file programmatically.

Then we start.

A chk file is a collection of several sections as follows.

```
struct ScenarioCHK {
  struct Section {
    uchar type[4];
    ulong len;
    uchar data[len];
  } section[?];
}
```

Actually, if expressed in C language, ScenarioCHK::Section::data part will be done with uchar [1] trick.

For example, let's start analyzing step by step. Again, we will start the analysis with (2)Bottleneck.scm.
Let's open scenario.chk with a hex editor. I have also put it in the attachments.

56 45 52 20 02 00 00 00 3B 00
VER ....;.
starts with .

56 45 52 20 02 00 00 00 3B 00
[type] [len] [data]

type = '56' '45' '52' '20' "TYPE"
len = 0x00000002 (2)
data = 3B 00

It's like this.

Read on.
49 56 45 52 02 00 00 00 0A 00
IVER.....
Read the same as above.

If you read like that, there will be items that become types. Let's record it all.

if you record
VER , IVER, IVE2, VCOD, IOWN, OWNR, ERA, DIM, SIDE, MTXM, PUNI, UPGR, ...
대충 이렇게 있을것에요. 이 각각 세션들의 설명은 위의 링크에서 보시면 됩니다.

0

---

scenario.chk를 분석하는 프로그램 chkparse를 만들어보도록 하겠습니다. chkparse의 역할은 다음과 같습니다.

1. 맵에 여러 단락들을 이름별로 분류해준다.
2. 원하는 단락을 바로 추출할 수 있도록 해준다.
3. 각 단락의 crc32값을 계산해준다. 언플 만들때 유용한데, 프로텍터가 어느 단락을 건드리는지 빠르게 확인할 수 있습니다.

chkparse의 알고리즘은 다음과 같습니다.

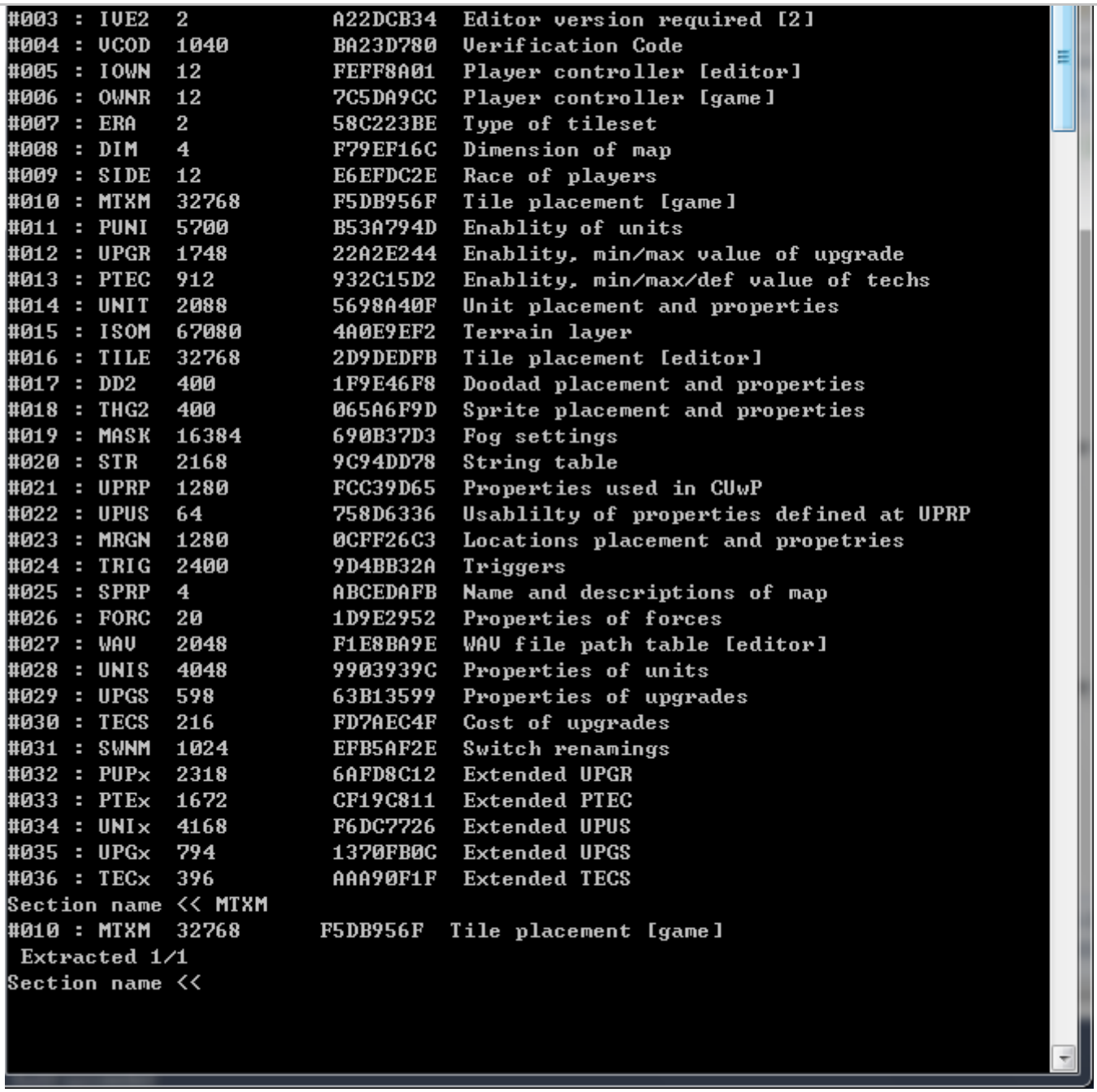
```
(파일 내용이 8byte 이상 남아있으면) {
   type와 len을 읽는다.
   len >= 0이라면 {
      파일 내용이 len 이상 남아있으면 {
         len만큼 읽는다.
      ;
   ;

   len < 0이라면 { // 주석 참조
      ftell < len : break;
      fseek(-len, SEEK_CUR);
   ;
;

//주석 : len < 0인 파일이 있어요; 제가 만들었거든요. 프로텍션에 쓰이는 한 기법 중 하나입니다. jsp라고도 하죠.
// 스타에서는 아무래도 type 자체를 인식하는가봅니다.
```

대충 이 알고리즘을 가지고 코드를 만들면 chkparse.zip과 같은 형태가 됩니다. scenario.chk를 적당히 넣고 실행시켜보면

0



좋아요. 이렇게 MTXM을 얻을 수 있게 됬어요. 굳굳굳.

이제 이 코드랑 전에 코드를 합치면 MTXM으로부터 타일 렌더링을 할 수 있겠죠?
이거는 쉬우니까 생략합니다. 연습으로 남겨둘께요.

P.S) Extended TECS같은 거는 각 섹션들의 설명입니다. rpgguild 자료에 나와있는 설명을 그냥 영어로 번역해본거예요.
... 조만간 고쳐야겠네요. rpgguild가 번역이 잘못된게 좀 있네요. 아무래도 SEN거를 기준으로 재번역해

0

scenario.chk
chkparse.zip

0

왜물어

whyask37님의 블로그입니다.

이웃추가

이 블로그 Project SMD 카테고리 글

6. 기초 작업

2013. 9. 12.

0

5. SFmpq (ShadowFlare's MPQ Library) 와 예제

2013. 9. 11.

1

4. scenario.chk

2013. 9. 10.

0

3. 지형 출력

2013. 9. 10.

3

0

whyask37's blog

이 블로그 인기글

MPQ 가지고놀기 (1) - 간단한 MPQ 파일 분석

2013. 10. 19.

11

5. SFmpq (ShadowFlare's MPQ Library) 와 예제

2013. 9. 11.

1

[뻘강의] 13. 트리거 프로그래밍 - TRIG-MRGN 루프

2014. 2. 24.

0

[뻘강의] 2. 데스 사이의 대입, 더하기

2014. 1. 19.

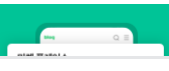
1

[뻘강의] 5. 포인터 예제 - 유닛 제한 줄이기

2014. 1. 23.

0

back to top



0

whyask37's blog

0