

# Map/Scenario XML Format For EDCE

Version 1.01

11/10/17

## Table of Contents

Map/Scenario XML Format For EDCE.....	1
Introduction.....	2
Map Data File Format.....	3
Scenario Data File Format.....	5

# ***Introduction***

The Map and Scenario Files, like most files used for interfacing with Empire Deluxe Combined Edition, use a psuedo xml format. (I say psuedo as it is not completely per-spec).

Update History -

v1.0

original first draft.

V1.01

added scenario format

# Map Data File Format

```
** File Begins **
<MDATA> //Map Data Header
  <MDA
    W="<width>" //map width
    H="<height>" //map height
    VW="<T/F>" //vertical wrap
    HW="<T/F>" //horizontal wrap
    NC="<nc>" //number of cities on the map
    SC="<sc>" //spawn point count
    IC="<ic>" > //important city count

    <DSV><![CDATA[YYYY.MM.DD H24:MM:SS]]></DSV> //Date (displayed only)
    <DSC><![CDATA[A Map]]></DSC> //Description
  </MDA>

  <MRS> //Map Rows, there will be Height * the entries

  //Each Map Row will have Width * characters in the string
  //Values Are (case important)
  // E = Edge
  // C = City
  // L = clear
  // S = Sea
  // A = Shallow
  // F = Forest
  // R = Rough/Desert
  // M = Mountain
  // V = River
  // H = Hill
  // W = Snow
  // P = Swamp
  // K = Peak
  <MR><![CDATA[EEEEEEEEEEEEEEEEEEEE]]></MR>
  ... x height times
</MRS>

  <MSDS> //Map Extra Data (resources/wasteland/road/mines)
  //This member may be empty (<MSDS/>)

  <MSD
    X="<x loc>" //Location X Value
    Y="<y loc>" //Location Y Value
    S="<res count>" //Resource Count (0 for none)
    R="<T/F>" //Has A Road
    M="<T/F>" //Has A Mine
    W="<T/F>"
  /> //MSD Closed
  ... 0...n
</MSDS>
```

```

<CIS>      //Cities On The Map (all neutral)
  <CD      //City Data
    X="<xloc>" //Location X Value
    Y="<yloc>" //Location Y Value
    G="<id#>" //city unique id
    E="<eff#>" //efficiency
    I="<T/F>" //is an important city
    P="<T/F>" //is a spawn point
  >
    <N><![CDATA[CheBro]]></N> //City Name
    <S><![CDATA[NE]]></S> //Unit Key For Specialty "NE"=None
  </CD>
</CIS>

</MCDATA> //Map Data Header Close

** EOF **

```

# Scenario Data File Format

\*\* File Begins \*\*

```
<SCEN>
  <SCNA> //Scenario Attributes
    <DSV><![CDATA[YYYY.MM.DD H24:MM:SS]]></DSV> //Date (displayed only)

    <D><![CDATA[Scenario Description Text]]></D>

    <U><![CDATA[Unit Set Key]]></U> //Unit Set Key linked to the Scenario

  <SPLYIS> //Scenario Player Info
    <SPYLI
      BP="<buy point value>" //Buy Points For The Position
    >
      <PC //Player Color (all values 0-255)
        TR="<foreground red>"
        TG="<foreground green>"
        TB="<foreground blue>"
        BR="<background red>"
        BG="<background green>"
        BB="<background blue>"
        XR="<text red>"
        XG="<text green>"
        XB="<text blue>"
      /> //player color closed
    </SPYLI> //player position closed

    ... 0...n

  </SPLYIS> //Scenario Player Info Closed

</SCNA> //Scenario Attributes Closed

<MDATA> //Map Data Header
  ...
  Map Data As Described in Previous Section
  ...
</MDATA> //Map Data Header Close

<SCIS> //Scenario Specific City Data
  <SCNC //individual city info
    G="<id#>" //city id defined in map data
    O="<0 to number players - 1>" //position owner (p#-1), so
    //Player #1 is "0", #2 is "1"
    C="<T/F>" //is a capital flag
    S="<#supply>" //amount of supply to start
  />
  ... 0...n
</SCIS> //Scenario Specific City Data Closed

<UNHS> //Units with no host
  ... 0 to n scenario units (defined below, host id must be 0)
</UNHS> //Units with no host closed
```

```

<HOS>          //Units With a host
... 0 to n scenario units (defined below, host id must not be 0, and
                        already defined in map data cities <CIS> or
                        scenario data not hosted units <UNHS> )
</HOS>         //Units With a host closed

<SLEXS>        //Scenario Location Extra Data List

    <SLEX          //Scenario Loc Extra Data
        X="<xloc>" //Location x coordinate
        Y="<yloc>" //Location y coordinate
        V="<0 to number players - 1>" //loc is visible to
                                        //position (p#-1), so
                                        //Player #1 is "0", #2 is "1"
    //>
... 0 to n
</SLEXS>       //Scenario Location Extra Data List

</SCEN>        //Scenario Data Header Closed
** EOF **

```

### Scenario Unit Definition

```

<SCNU          //Scenario Unit
    G="<id#>" //unique unit id
    O="<0 to number players - 1>" //position owner (p#-1), so
                                    //Player #1 is "0", #2 is "1"
    H="<id#>" //unique host unit id,
                                    //or 0 for none
    U="<unit type key>" //Unit Set unit type Key
    X="<xloc>" //Location x coordinate
    Y="<yloc>" //Location y coordinate
    L="0" //value must be 0
    V="<level value>" //Level Value Code
                    //L = any ground,
                    //O = orbital
                    //U = sub (deep)
                    //Note: if hosted, must have
                    //same level as host, cities
                    //use level "L" (ground)
>
    <N><![CDATA[Unit Name]]></N> //unit's name

</SCNU>       //Scenario Unit Closed

```

Note on unique ids:

City And Unit ids come from the same set, so they must unique. You cannot have a city and a unit share the same id. It is also important to start at 1, and work up incrementally. Unique id value 0 has meaning (not an existing unit).

Note on player positions:

Player positions are indexed starting with 0 == Player position #1.

For scenarios, a unit that has a range is supposed to start within a host. I would say the results of starting one outside of one is not "expected" and may not function as desired.