ZunTzu Manual

Table Of Contents

How to Create a Game Box

Game boxes are stored as .ztb files. This article explains how to create your own game box file from scratch. Actually it's probably easier to create a game box modifying an existing one. Although this article may seem technical, creating a game box is actually very easy; it is basically just a matter of scanning the board and the counter-sheet, and writing a small text file describing how many dice are required and how to punch the counters.

File Types

game = .ztg
scenario = .zts
game-box = .ztb

How to Create the Game Box Archive

You can create the game-box.xml file with WordPad or your favorite text editor. Just make sure you save the file in plain text format (.txt) and then rename it with a .xml extension. Technically savvy readers will find a formal description of the structure of the file here.

The example below shows a simple game-box.xml file.

```
<?xml version="1.0" encoding="windows-1252" ?>
<game-box
        name="Paris vaut bien une messe"
        description="Encart-jeu VaeVictis n 50, les guerres de religion 1562-1598."
        copyright="VaeVictis – Histoire &amp; Collections 2003"
        startup-scenario="Dreux.zts">
</game-box>
```

Indicate the name of your game box, a description and who owns the copyright for the original game. Then, indicate the scenario file that will be loaded on startup. Don't worry if you know nothing about XML, just copy the example and replace the text content. All you need to know is that characters "&", "<" and ">" must be replaced respectively by "&", "<" and ">".

In the same archive as game-box.xml, you can provide an image file that will be used as an icon in the game library. The icon image should be sized 48x48 and the file must be named "icon.bmp".
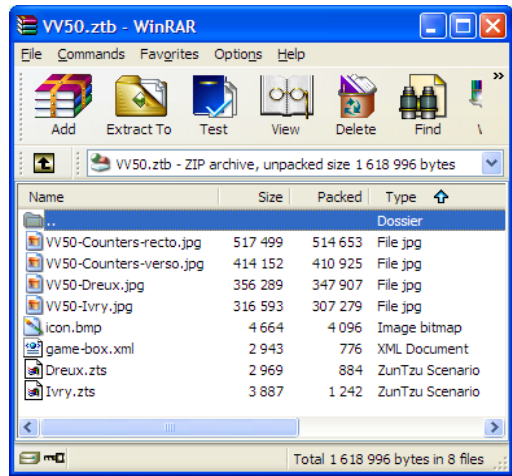
The content of a typical game box.

How to Add Dice

All the classic types of dice are supported, four-sided "D4", six-sided "D6", eight-sided "D8", ten-sided "D10", twelve-sided "D12" and twenty-sided "D20". Group dice can be cast together in the same dice hand. You can define several dice hands of differently colored dice.

Animated 3D dice.

Below is an example of two dice hands of two ten-sided dice.

```xml
<?xml version="1.0" encoding="windows-1252" ?>
    <game-box
        name="Paris vaut bien une messe"
        description="Encart-jeu VaeVictis n°50, les guerres de religion 1562-1598."
        copyright="VaeVictis - Histoire &amp; Collections 2003"
        startup-scenario="Dreux.zts">
    <dice-hand type="D10">
        <dice count="2" color="ffffff" pips="000000" />
    </dice-hand>
    <dice-hand type="D10">
        <dice color="0000a0" pips="ffffff" />
        <dice color="a00000" pips="ffffff" />
    </dice-hand>
    <map name="Ivry" image-file="VV50-Ivry.jpg" resolution="150 dpi" />
    <map name="Dreux" image-file="VV50-Dreux.jpg" resolution="150 dpi" />
```

```
<counter-sheet name="Planche de pions"
        front-image-file="VV50-Counters-recto.jpg" front-resolution="150 dpi"
        back-image-file="VV50-Counters-verso.jpg" back-resolution="150 dpi">
</counter-sheet>
</game-box>
```

The color value is the color of the dice in hexadecimal code. Most bitmap graphics editors can give you the hexadecimal code of a color. Use "000000" for black, "ffffff" for white, "ff0000" for red, "00ff00" for green or "0000ff" for blue. The pips value is the color of the dots or numbers on the dice.

Dice notes

All the dice must have the same number of sides (all D6's for instance) but you can mix dice with different colors or textures into a single hand.

```
<dice-hand type="D6">
  <dice color="0000a0" pips="ffffff" />
  <dice color="a00000" pips="ffffff" />
  <dice texture-file="dicetexture1.png" />
  <dice texture-file="dicetexture2.png" />
</dice-hand>
```

How to Add maps

This is the most time consuming and most delicate step, especially if your game board is huge. ZunTzu supports three scan resolutions: 150, 300 and 600 dpi. 150 dpi should be fine for most boards.

Because your scanner is likely to be smaller than the board, you'll have to scan the board part by part, and build a composite image using a bitmap graphics editor (such as the GIMP or Adobe Photoshop). Save the composite image as a high quality JPEG file (PNG is not recommended because of the noise introduced by the scanning.

Then update your game-box.xml file. The example below shows a game box with two maps.

```
<?xml version+"1.0" encoding="windows-1252" ?>
<game-box
        name="Paris vaut bien une messe"
        description="Encart-jeu VaeVictis n50, les guerres de religion 1562-1598."
        copyright="VaeVictis – Histoire &amp; Collections 2003"
        startup-scenario="Dreux.zts">
      <map name="Ivry" image-file="vv50-Ivry.jpg" resolution="150 dpi" />
      <map name="Dreux" image-file="vv50-Dreux.jpg" resolution="150 dpi" />
</game-box>
```

ZunTzu will display the name of the maps in the tabs at the bottom of the screen.

How to Add Counter Sheets

It's a lot easier if your game counters are still unpunched, scan the counter sheet in the same way as you scanned the board. If your game has already been used and the counters punched, you'll have to reconstitute the counter sheet somehow. Most counter sheets will look good when scanned at 150 dpi. If the counters contain text or numbers in small print consider using a higher resolution like 300 dpi. Maps and counter sheets don't have to be all scanned at the same resolution. Most scanners will cast a shadow on the counters as the light source moves across the glass pane. It is better if that shadow is consistent with the lighting conventions in Windows, the top and left edge of the counters should be lighter than the bottom and right edge. You can achieve that by choosing the correct orientation for the counter sheet on the scanner glass pane.

Note the brighter left edge, giving a nice 3D look.

If the counters are single-sided, it's OK to scan only the front of the counter sheet, otherwise scan both sides. Save each side as a separate JPEG file. Update your game-box.xml file. The example below shows a game box with a single counter sheet.

```
<?xml version="1.0" encoding="windows-1252" ?>
<game-box
        name="Paris vaut bien une messe"
        description="Encart-jeu VaeVictis n°50, les guerres de religion 1562-1598."
        copyright="VaeVictis - Histoire &amp; Collections 2003"
        startup-scenario="Dreux.zts">
        <map name="Ivry" image-file="VV50-Ivry.jpg" resolution="150 dpi" />
        <map name="Dreux" image-file="VV50-Dreux.jpg" resolution="150 dpi" />
        <counter-sheet name="Planche de pions"
                front-image-file="VV50-Counters-recto.jpg" front-resolution="150 dpi"
                back-image-file="VV50-Counters-verso.jpg" back-resolution="150 dpi">
        </counter-sheet>
</game-box>
```

ZunTzu needs to know how to punch the counters off the counter sheet. You don't actually have to indicated the location of each counter on the counter sheet, you just have to indicate the location of counter sections. A counter section is a rectangular area that contains rows of counters placed side-by-side. Indicate the coordinates in pixels of the top left corner and the right bottom corner of the counter section, the number of rows and the number of columns.

A counter section of three rows of six counters.

If the counters are double-sided, provide different coordinates for the front and back sides. Below is an example, with double-sided, front-sided only and back-side only counter sections. Counter sections of one row and one column are used to describe individual counters (actually you don't have to provide a value for "rows" and "columns" in that particular case).

```xml
<?xml version="1.0" encoding="windows-1252" ?>
<game-box
        name="Paris vaut bien une messe"
        description="Encart-jeu VaeVictis n°50, les guerres de religion 1562-1598."
        copyright="VaeVictis - Histoire &amp; Collections 2003"
        startup-scenario="Dreux.zts">
        <map name="Ivry" image-file="VV50-Ivry.jpg" resolution="150 dpi" />
        <map name="Dreux" image-file="VV50-Dreux.jpg" resolution="150 dpi" />
        <counter-sheet name="Planche de pions"
                front-image-file="VV50-Counters-recto.jpg" front-resolution="150 dpi"
                back-image-file="VV50-Counters-verso.jpg" back-resolution="150 dpi">
                <counter-section rows="10" columns="6"
                        front-left="16" front-top="10" front-right="550" front-bottom="903"
                        back-left="581" back-top="13" back-right="1112" back-bottom="907" />
                <counter-section rows="10" columns="6"
                        front-left="564" front-top="10" front-right="1099" front-bottom="904"
                        back-left="33" back-top="8" back-right="564" back-bottom="904" />
                <counter-section rows="1" columns="3"
                        front-left="833" front-top="904" front-right="1099" front-bottom="993"
                        back-left="33" back-top="902" back-right="298" back-bottom="992" />
                <counter-section
                        front-left="460" front-top="938" front-right="549" front-bottom="1028"
                        back-left="580" back-top="940" back-right="669" back-bottom="1029" />
                <counter-section
                        front-left="566" front-top="938" front-right="656" front-bottom="1029"
                        back-left="474" back-top="939" back-right="562" back-bottom="1028" />
                <counter-section rows="7" columns="5"
                        front-left="106" front-top="1028" front-right="550" front-bottom="1653"
                        back-left="580" back-top="1028" back-right="1023" back-bottom="1655" />
                <counter-section rows="7" columns="6"
                        front-left="567" front-top="1029" front-right="1099" front-bottom="1654"
                        back-left="32" back-top="1025" back-right="563" back-bottom="1653" />
                <counter-section
                        front-left="19" front-top="1563" front-right="107" front-bottom="1653"
                        back-left="1023" back-top="1566" back-right="1111" back-bottom="1655" />
```

```
        <counter-section
                front-left="334" front-top="922" front-right="424" front-bottom="1013" />
        <counter-section rows="6" columns="1"
                front-left="9" front-top="1022" front-right="98" front-bottom="1556" />
        <counter-section rows="1" columns="2"
                back-left="693" back-top="921" back-right="870" back-bottom="1010" />
        <counter-section rows="6" columns="1"
                back-left="1032" back-top="1026" back-right="1120" back-bottom="1560" />
    </counter-sheet>
</game-box>
```

Counter Notes

Yes, on the same counter sheet you can define a counter-section and a card-section.

With the four coordinates you determine the rectangle of the area where your counters are in that graphic. By using the rows and columns attributes you can have ZunTzu automatically split up that area in equal division by the number of rows (vertically) and columns (horizontally).

Example you counter.png has six counters on it, from 0,0 to 300,200 (no border around it, just counter). You could do each counter individually, like;

```
<counter-section rows="1" columns="1"
        front-left="0" front-top="0" front-right="100" front-bottom="100" />
<counter-section rows="1" columns="1"
        front-left="100" front-top="0" front-right="200" front-bottom="100" />
<counter-section rows="1" columns="1"
        front-left="200" front-top="0" front-right="300" front-bottom="100" />
<counter-section rows="1" columns="1"
        front-left="0" front-top="100" front-right="100" front-bottom="200" />
<counter-section rows="1" columns="1"
        front-left="100" front-top="100" front-right="200" front-bottom="200" />
<counter-section rows="1" columns="1"
        front-left="200" front-top="100" front-right="300" front-bottom="200" />
```

But that is just a lot of work. If you say;

```
<counter-section rows="2" columns="3"
        front-left="0" front-top="0" front-right="300" front-bottom="200" />
```

Now you have replaced those 6 items with just 1, and ZunTzu will do the cutting for you. It will create 2*3 = 6 counters of equal size; width 300 / columns = 100 wide; height 200 / rows –= 100 tall. You only say rows and columns once since you always create 1 background for each 1 foreground (never an unequal number). So if you create 6 front counters, you create 6 back counters as well. ZunTzu will do the same cutting for you on your background counters image, and match up the images.

Foreground image
A B C
D E F
Background image
c b a
f e d

Another example

```
<counter-sheet name="Blips"
              front-image-file="BlipsFront.png" front-resolution="300 dpi"
              front-mask-file="BlipsFrontMask.png"
              back-image-file="BlipsBack.png" back-resolution="300 dpi"
              back-mask-file="BlipsBackMask.png">
    <counter-section rows="3" columns="9"
              front-left="0" front-top="1" front-right="1350" front-bottom="450"
              back-left="0" back-top="1" back-right="1350" back-bottom="450" />
</counter-sheet>
```

Counter Wheels

Here is a workaround for dice used as markers, some miniatures gamers already use counters shaped like numbered wheels. You can decrement or increment the counter by rotating the wheel. It's even more convenient than the real thing because you don't have to look for the face you want.
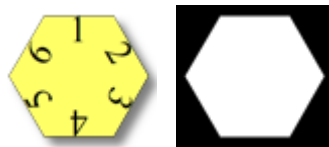
Here is an example and the mask

To read the value of such a counter, just look for the number at the top. You can increment or decrement the value by rotating the counter. You would setup one such counter on each hex at the start of the game. Each time you stack a tile, you will increment the counters overlapping that tile by rotating the wheels. For a three space tile, you would have to rotate three counters.

Drawing Counters at Random

To emulate the random picking of counters from a cup, organize all your counters faces down in a single stack, then double click on the stack to open the stack inspector and shuffle the stack with the "Shuffle" function (indicated by this icon Shuffle). Each time you need to draw a counter, remove the counter at the top of the stack and reveal it.

Supply

Is it possible to add some kind of counters, like markers in cyberboard, to have an unlimited number of them: as numbers, battle markers, arrows etc. You can't specify an unlimited number, only an arbitrarily large number. Insert this in each counter section definition, supply-"100"

```
</counter-sheet>
  <counter-sheet name="Chips" front-image-file="chips.jpg" front-mask-file="chips-mask.png">
    <counter-section front-left="505" front-top="506" front-right="807" front-bottom="808" />
```

```
    <counter-section columns="4" front-left="0" front-top="0" front-right="984" front-bottom="246"
supply="100" shadow="40" />
    <counter-section columns="4" front-left="0" front-top="246" front-right="984" front-bottom="492"
supply="50" shadow="40" />
    <counter-section columns="2" front-left="0" front-top="492" front-right="492" front-bottom="738"
supply="20" shadow="40" />
    <counter-section columns="2" front-left="0" front-top="738" front-right="492" front-bottom="984"
supply="10" shadow="40" />
  </counter-sheet>
```

Shadows

The shadow attribute just affects how far each counter casts a shadow. To disable the drop
shadow for the counters just put shadow="0" at the end of each counter section declaration
(before the end of line "/>" characters). The default shadows attribute value for counters is "20",
and cards it is "5". You can have shadows on terrain, default is "0", setting it a value other than
"0" casts shadows on terrain pieces as well.

How to Handle Arbitrary Shaped Counters

If the counters are not rectangular, you must provide a transparency mask image to indicate the
edges of the counters.
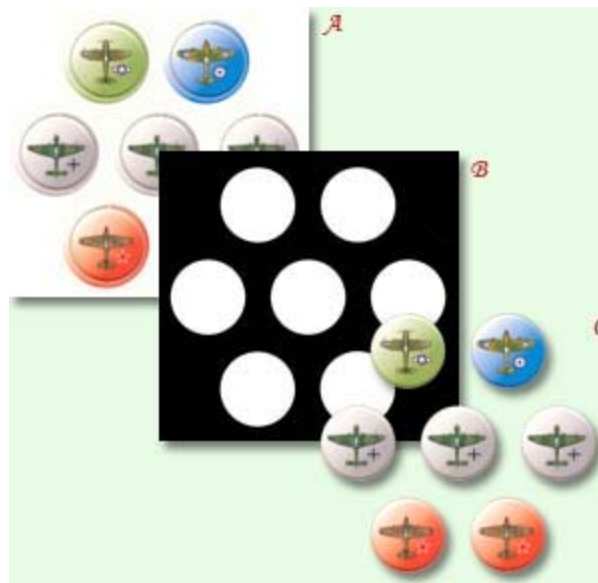
A mask is used to cut off round counters.
A: image, B: mask, C: result

A transparency mask image is simply a grayscale image of the same size and resolution as the
counter sheet image. Paint the areas inside counters white and the areas outside counters
black. Use intermediate gray levels to draw nice antialiased edges. Save your image as a PNG
file, as a lossless compression scheme, it is better file format than JPEG for mask files. If the
counters are double-sided, provide a different mask for the front side and the backside. Below is
an example of a double-sided counter sheet with masks.

```
<counter-sheet name="Planche de pions"
        front-image-file="VV50-Counters-recto.jpg" front-resolution="150 dpi"
        front-mask-file="VV50-Counters-recto-mask.png"
        back-image-file="VV50-Counters-verso.jpg" back-resolution="150 dpi"
        back-mask-file="VV50-Counters-verso-mask.png">
```

Cards

You can unfold a card stack by holding the Control key. You can move a stack of "cards" by pressing "Cntrl" and selecting the bottom one in the stack.

```
<?xml version="1.0" encoding="windows-1252" ?>
<game-box
        name="Poker (v2)"
        description="Poker set, with cards and chips."
        copyright="Cards design based on &quot;KEM Arrow&quot; by the U.S. Playing Card Co."
        startup-scenario="startup.zts">
        <map name="Table" image-file="table.jpg" />
        <counter-sheet name="Cards" front-image-file="cards.jpg" front-mask-file="cards-mask.png">
                <card-section rows="5" columns="8"
                        face-left="0" face-top="0" face-right="3070" face-bottom="2665"
                        back-left="2686" back-top="2665" back-right="3070" back-bottom="3198" />
                <card-section rows="2" columns="7"
                        face-left="0" face-top="2665" face-right="2686" face-bottom="3731"
                        back-left="2686" back-top="2665" back-right="3070" back-bottom="3198" />
                <card-section rows="5" columns="8"
                        face-left="0" face-top="0" face-right="3070" face-bottom="2665"
                        back-left="2686" back-top="3198" back-right="3070" back-bottom="3731" />
                <card-section rows="2" columns="7"
                        face-left="0" face-top="2665" face-right="2686" face-bottom="3731"
                        back-left="2686" back-top="3198" back-right="3070" back-bottom="3731" />
        </counter-sheet>
        <counter-sheet name="Chips" front-image-file="chips.jpg" front-mask-file="chips-mask.png">
                <counter-section front-left="505" front-top="506" front-right="807" front-bottom="808" />
                <counter-section columns="4" front-left="0" front-top="0" front-right="984" front-bottom="246" supply="100" shadow="40" />
                <counter-section columns="4" front-left="0" front-top="246" front-right="984" front-bottom="492" supply="50" shadow="40" />
                <counter-section columns="2" front-left="0" front-top="492" front-right="492" front-bottom="738" supply="20" shadow="40" />
                <counter-section columns="2" front-left="0" front-top="738" front-right="492" front-bottom="984" supply="10" shadow="40" />
        </counter-sheet>
</game-box>
```

Terrains

The most important new feature is the terrains. Terrains sheets are defined just like counter sheets, except you need to replace "counter-sheet" with "terrain-sheet" and "counter-section" with "terrain-section". Terrain sheets don't have tabs except when you switch to terrain editing mode (with the trowel icon).

```
<terrain-sheet name="Stock Price Index" front-image-file="SPI.png" front-resolution="150 dpi" >
    <terrain-section front-left="1" front-top="1" front-right="1611" front-bottom="1229"/>
</terrain-sheet>
```

Terrains will be moveable and clone-able. They will be displayed beneath cards and counters. Counters become invisible when terrain editing mode is on. It can be convenient to check map features underneath stacks of counters even if you don't use terrains.

Empty Map for Terrain Placement

Some games will use terrains as the only background elements. It is thus possible to define an empty black map, not linked to any image file, to display the terrain elements. In the scenario file, just declare a "layout" node with a "tab" attribute for the tab text, but no "board" attribute.

Terrains Possibilities

A scenario designer combines several geomorphic maps to create a scenario for a WWII tactical wargame. Player sets woods, rivers and hills out on the table before a miniature game. A game master reveals a room at the end of a corridor during a live roleplaying game session.

How to Add Scenarios

A scenario file enumerates the maps and counter sheets that will be used and the initial placement of each counter. You must provide at least a startup scenario in your game box archive. Scenarios are actually regular saved game files renamed as .zts files (instead of .ztg). The easiest way to create a scenario is to begin with a simple dummy scenario with all the counters unpunched, and then run ZunTzu, move the counters to their starting locations, then save the game. All you'll have to do then is replace the dummy startup scenario by the saved game renamed as a .zts file. Below is an example of a simple dummy scenario;

```
<?xml version="1.0" encoding="Windows-1252"?>
<game
        game-box="Paris vaut bien une messe"
        scenario-name="Dreux"
        scenario-description="Dreux, 19 Décembre 1562"
        scenario-copyright="VaeVictis - Histoire &amp; Collections 2003">
        <layout board="Dreux" />
        <layout board="Planche de pions" />
</game>
```

The value for "game-box" should be the same name as the one used in the game-box.xml file. A scenario can use all the maps and counter sheets, or just some of them. The above scenario file indicates that two boards will be used, "Dreux" and "Plance de pions".

Creating the Scenario file

To create the final scenario file from your dummy scenario, follow this procedure

1. Pack all your files in a zip archive.
2. Rename the archive with a .ztb extension.
3. Double-click the .ztb file to launch ZunTzu. In ZunTzu, move the counters, resize the views, then save the game.
4. Quit ZunTzu, replace the dummy scenario file in the game box archive by the saved game.

Scenario notes

Scenario Order

I would like to include several scenarios in my ZunTzu Gamebox. They should appear in numeric order (01 Scenario 1, 02 Scenario 2, etc.). In which way are they sorted when you try to open them? Do I have to rename them in a special way? They appear in the order that you put them in the .zip folder.

Dummy Scenario

The purpose of the dummy scenario file is simply to indicate which maps and counter sheets will be used through "layout" nodes. Then you will be able to load the game box in ZunTzu and set the counters out before saving the game file that will replace your dummy file.

Start Scenario

It is best to keep the counter sheet on a separate tab. It means you'll have to move the counters from the counter sheet tab to the game board. You can speed up the setup process by creating a scenario with all the counters already on the board, ready to play. Just start with a simple scenario file, with no "stack" tags, just "layout" tags to indicate which boards to use. Then launch ZunTzu, move the counters to their initial positions, the save game and close ZunTzu. Now all you have to do is rename the saved game with a ".zts" extension and replace your simple scenario file with that new file.

Tabs

Note that you can also use the "tab" attribute with regular "layout" nodes. It can be used to implement two tabs with the same image. It can also be used to translate the tab text (at the scenario level).

Regarding the 'tab' feature

Normally, when you built up a scenario for ZunTzu, you start with a very simple XML file with statements such as this one;

```
<layout board="My board" />
```

The code above creates a tab named "My board" that contains the graphics for the "My board" map (as defined in the game-box.xml file).

Change Tab Name

The undocumented 'tab' feature allows you to change the name of the tab;

```
<layout board="Off-board chart" tab="Red chart" />
<layout board="Off-board chart" tab="Blue chart"/>
```

The code above creates two tabs named "Red chart" and "Blue chart" that both contains the same graphics for the "Off-board chart" map (as defined in the game-box.xml file).

Translate a Tab

You can use the 'tab' feature to translate a tab. For instance;

```
<layout board="Main board" tab="Plateau principal" />
```

That's the only textual information in a game, apart from possible text in the graphics.

Create an Empty Tab

You can also use the 'tab' feature to create an empty tab (a tab without graphics, all black);

```
<layout tab="Main board" />
```

That's very useful with geomorphic boards, the main tab is usually empty because all the boards are defined as terrains.

Bill Barrett's notes

Ok guys, I've just discovered that ZunTzu has a whole lot more power under the hood than Jerome has admitted!

For months now, I've been intrigued by these two sentences:

"Note that you can also use the "tab" attribute with regular "layout" nodes. It can be used to implement two tabs with the same image."

I thought they could be useful, but couldn't figure out how. Then suddenly, in a "Eureka" moment, it hit me...

You can have two instances of a game running, in parallel, in one gamebox session!

All you have to do is add one line to a Scenario file - something like:

<layout board="Map" tab="Game 2" />

where "Map" equals whatever you've called your main board in the game-box file.

So I added it to my France 1940 gamebox, loaded it into ZunTzu, and sure enough tab "Game 2" displayed an exact copy of the map. So then I thought what the heck - let's try six games at once...

and I added these further lines:

<layout board="Map" tab="Game 3" />
<layout board="Map" tab="Game 4" />
<layout board="Map" tab="Game 5" />
<layout board="Map" tab="Game 6" />

Loaded it up and sure enough, six separate tabs, all with copies of the map in them.

Now I had originally "supplied" six German and eleven Allied "counter-sheets" to this gamebox to provide enough pieces for players to chose from the 66 possible combinations of setup, so it was a simple matter to use these extra "counter-sheets" to set up six separate games.

It took a while (but I had to see if it worked and also to see how ZunTzu would perform), and when I finished I saved it as a separate .ztg file.

When I loaded it back up, I noticed it was slow (it took about 2 minutes longer than the single game scenario) and also that the panning/zooming was a little jerky.

But remember, this has a fairly large 300dpi map and there would be about six hundred 300dpi counters to position on the maps (and six games at once is maybe a little excessive). However, if you turn down ZunTzu's display settings (and/or if you have more memory/a better processor) it's as smooth as ever. Also it seemed that performance improved once ZunTzu had been running for a while.

OK, so that's pretty cool, but it might not always be a good idea to use the "supply" attribute to increase the numbers of counters in every gamebox (some games use the physical limit of counters supplied as the Force pool for example) and you'd never know if anyone wanted to play 12 games at once (say) so what would be your limit? This is where it gets even better:

Out of curiosity (I didn't think it'd work for a minute!) I added the line:

<layout board="Counters" tab="Counter Sheet 2" />

Saved it, loaded it, and to my amazement, there on the second tab, was a complete fully functioning copy of the Counter sheet! It was identical in every way, even down to the individual "supply" numbers of each counter). Even more impressive was that when you move one of them to a map, and press the delete key, it "knows" which tab it belongs to and returns there.

So then of course, I tried it with cards... and exactly the same thing happened!

And of course Terrains are "unlimited" out of the box from the start, so you don't need to do anything as far as they're concerned.

Right, on the basis that a demonstration is worth a thousand words I suggest you do this: download a copy of Jerome's Poker gamebox, open it up, copy the "startup.zts file" to somewhere else and load it into your favorite text editor. Go to the bottom of the page and paste (make sure you do this on the line just ABOVE the "</game>" command):

```
<layout board="Table" tab="Table 2" />
<layout board="Cards" tab="Cards 2" />
<layout board="Chips" tab="Chips 2" />
```

So that the last FOUR lines should look like this:

```
<layout board="Table" tab="Table 2" />
<layout board="Cards" tab="Cards 2" />
<layout board="Chips" tab="Chips 2" />
</game>
```

Save it as something like "2 games setup.zts", copy it into Poker-v2.zip, rename the folder to .ztb and load it into ZunTzu. On the scenario menu it will be the second one listed.

You'll see the power and scope of this - and remember, all you had to do was add three lines of text to a scenario file! So what are the possible uses?

1) For me personally, it would be great to have the opportunity of playing two games at the same time (take the Allied side in one and the Axis in the other for example).

2) It's dead easy to set up three maps and counter sets to deal with hidden placement. You might use this code:

```
<layout board="Map" tab="Hidden Map A" />
<layout board="Map" tab="Hidden Map B" />
<layout board="Counters" tab="Hidden Counters A" />
<layout board="Counters" tab="Hidden Counters B" />
```

3) Rather than have each scenario in a gamebox load one at a time, you could set up a "scenario" which had all of them loaded simultaneously, each with their own tab.

4) It's a brilliant play testing tool - imagine being able to have six games setup to explore six different opening turn strategies. Or, you could test a game with optional rules/pieces against one without them...

And I'm sure there are others I haven't yet thought of…

Obviously there are a couple of caveats:

You've got to be careful that you don't mix up which game tab you're currently concentrating on – it'd obviously be easy to move cards/counters into one in which they don't belong. But the biggest problem is that you only have one player hand. If you're playing multiple sessions of games with cards, each player will have to "empty" their hands out onto the "table" before moving onto the next – irritating maybe, but not a fatal flaw.

Hiding Boards

Another feature is the possibility to hide a board to other players by clicking on the eye icon. It can be used to emulate off-board charts like the strategic charts in the Victory Games Fleet series or the off-board charts in the Kevin Zucker games. Use the "Hide board" icon to hide an off-board map from the other players. They'll see a black screen instead of the secret board. There is currently no player ownership of game pieces nor the ability to hide pieces from other players. On the other hand, you can own a board and hide it from the other players.

Players Hand

You can now increase a little more the height of the player's hand, and cards pop up from the hand when you move the cursor over them (easier reading). You can alter the size of the cards in your hand, the same way you zoom in and out on the main area. Hold down the right mouse button over the hand window and move the mouse forwards to zoom in (make cards larger) move mouse backwards to zoom out (make cards smaller). The player hand is a great way to store counters and move them from tab to tab but other players are not allowed to examine its content. Allocate some space on the board where the players will display the content of their 'hands'.

Space bar

Space bar shortcut to easily bring counters and cards into your hand.

Stacks

Moving counters from counter sheet to map. I make a stack and then move that stack to the other map, its very simple. (have to try this?) I ended up hard-coding the startup stack positions into my scenario templates, so now they all look neat and tidy, ready for me to build a new mission.

Stacking Space Between Counters

It seems that the resolution of your scan is too low. ZunTzu assumed that the images were 150 dpi when in reality they were probably closer to 80 dpi. That's why there seems to be large offsets between the counters. Actually the offsets are OK, it's the counters that are tiny. You are correct Jerome. I changed the dpi setting for counters and map to 150 and the stacks became somewhat tighter.

Shuffle

To emulate the random drawing of Activation Markers from a cup, organize all markers faces down in a single stack then shuffle the stack with the "Shuffle" function (indicated by this icon in the stack inspector). Each time you need to draw a marker, remove the marker at the top of the stack and reveal it.

What Needs to be Explaind

- how to use playing cards during game
- how to use terrains during game
- how to design a game box with cards
- how to design a game box with terrains (as tiles or geomorphic maps)
- how to design dice with custom textures

- the differences between counters, cards and terrain from a designer's point of view

14th of January 2008
So... what's new in v1.2?

ZunTzu 1.2 is out, offering terrains tiles, overlays and geomorphic maps, private off-map displays, enhanced experience, through features such as a space bar shortcut to easily bring counters and cards into your hand, support for more languages (Italian and Portuguese), sound effects, better error diagnostics for game box authors.

Other features are meant to enhance the user experience:

- long-awaited diagnostics for game box errors when using the Add button in the library dialog box. It will bring some relief to game box authors.

- notification message boxes when ZunTzu detects that a firewall (or other network device) is getting in the way.

- faster loading of games when texture compression is on. And the quality is better too. Give it a try by loading Africa II.

- scrolling buttons to help with games with a lot of tabs. You're not limited to using short tab names anymore.

- spatial separation of voices in games with more than two players, by using 3D sound buffers.

subtle sound effects for deck shuffling and dice casting. A (less subtle) sound notification when a player joins.

great keyboard shortcuts: Up and Down arrows to rotate counters, and Space to bring counters into a player's hand (very convenient).

Display Order

A summary of the different kinds of items in ZunTzu, in display order (from bottom-most to topmost):

Boards (maps and counter sheets): the board is displayed below everything else.
Unpunched counters and cards: displayed just above the board.
Terrains ("locked" tiles, geomorphic maps, overlays, ...)
Cards (punched)
Counters (punched): displayed above cards.
Everything else: dice, cursors, ...

What exactly are unpunched counters? How do we use/declare them?
Nothing new. It's just how I call counters that are still attached to their counter sheets.

Resolution

ZunTzu supports only 150, 300 and 600 dpi.

Changing scale

To change the scale of the counters relative to the board, just lie to ZunTzu: tell it that boards and counters are scanned at 150 dpi, but scan the boards at 150 x 125% = 187.5 dpi. Actually, I always scan at 600 dpi, because it's the optimal resolution of my scanner, and then I downscale the images in a bitmap graphics editor to 25% to get 150 dpi. Just do the same, but downscale the boards to 25 * 125% = 31.25 %.

A piece of advice: using Paint.NET, apply a slight blur before downscaling the 600 dpi image, just enough to hide the halftone screen used by the printing press. The image will look better and will be better prepared for JPEG compression. If the image is still a bit blurry after dowscaling, apply a sharpen effect as a last stage.

You should not trust the 72 dpi value. It is often just a default value used by bitmap graphics editors when the value is undetermined. Use your best judgement to evaluate the relative scale of each image, then put everything to the same scale by downsizing the images with the highest definition, and declare everything as 150 dpi in ZunTzu. If you have high definition scans for the counters, you can save them at double definition - relative to the maps - and then declare them as 300 dpi in ZunTzu.

Adding Comments

Syntax

If possible, what is the syntax to add comments to the file?

I tried blank lines, looks like its OK.

I also looked into another game-box and looks like comments are done like this....

<!-- this is a comment -->

Are the dashes needed?

Yes, the dashes are a must (and you have to use the start and end comment tags).

One of the quick easy ways I check my XML syntax is to open the xml file in my browser (IE7) which will format the file and verify the XML syntax. This helps me catch missing "/>" versus ">" markup.

Structure of a Game Box XML File

Below is a formal definition of the structure of a game-box.xml file.

A game-box.xml file is composed of a header and a body.

The header is always the same (although you can change the "encoding" attribute in case you want to use a non-latin alphabet).

Some attributes are optional. When attributes are missing Zuntzu will use these default values:

    The default value of "icon" is "icon.bmp".
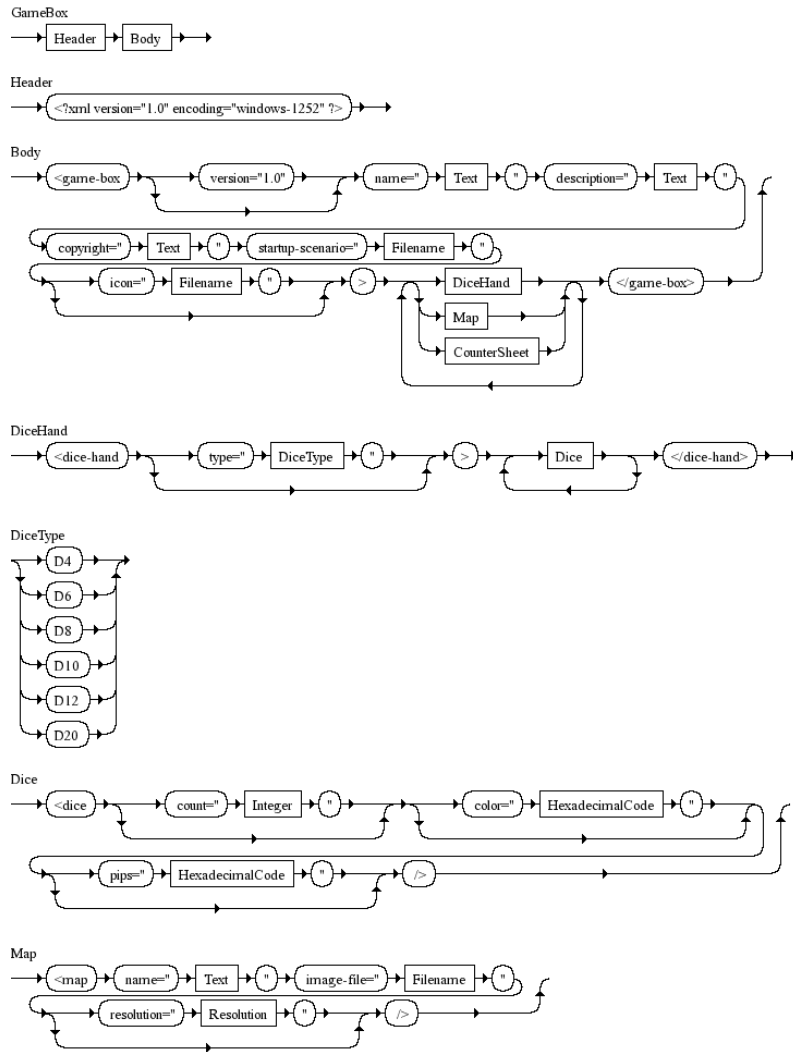   The default value of "dice-hand type" is "D6".
    The default value of "dice count" is "1".
    The default value of "dice color" is "ffffff" (white).
    The default value of "dice pips" is "000000" (black).
    The default value of "resolution" is "150 dpi".

GameBox

Header → Body

Header

<?xml version="1.0" encoding="windows-1252" ?>

Body

<game-box version="1.0" name=" Text " description=" Text "
copyright=" Text " startup-scenario=" Filename "
icon=" Filename " > DiceHand / Map / CounterSheet </game-box>

DiceHand

<dice-hand type=" DiceType " > Dice </dice-hand>

DiceType

D4
D6
D8
D10
D12
D20

Dice

<dice count=" Integer " color=" HexadecimalCode "
pips=" HexadecimalCode " />

Map

<map name=" Text " image-file=" Filename "
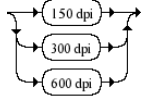resolution=" Resolution " />

Attribute "counter-section type" is deprecated and shouldn't be used. It exists only for backward compatibility.

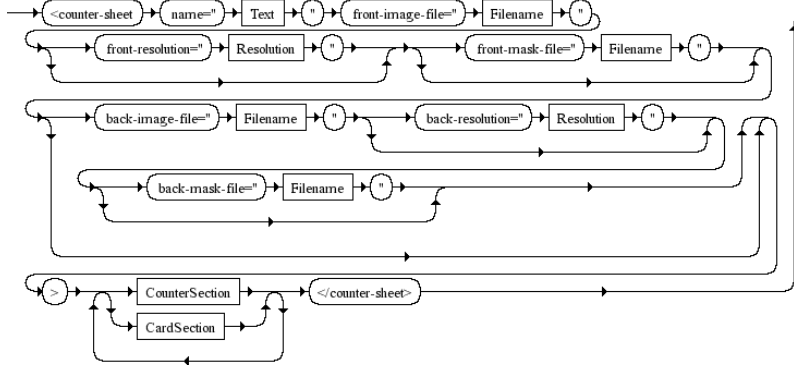The default value of "rows" and "columns" is "1".
The default value of "supply" is "1".
The default value of "shadow" is "20.0" for "counter-section" and "5.0" for "card-section".
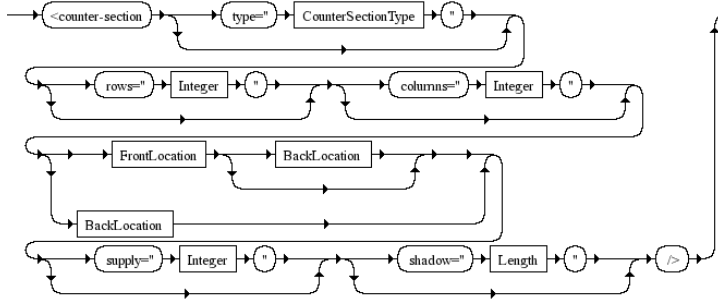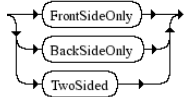
Resolution

- 150 dpi
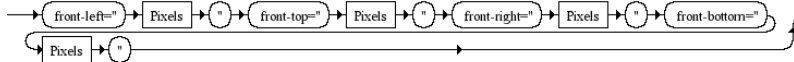- 300 dpi
- 600 dpi

CounterSheet

`<counter-sheet` `name="` Text `"` `front-image-file="` Filename `"`
`front-resolution="` Resolution `"` `front-mask-file="` Filename `"`
`back-image-file="` Filename `"` `back-resolution="` Resolution `"`
`back-mask-file="` Filename `"`
`>` CounterSection / CardSection `</counter-sheet>`

CounterSection

`<counter-section` `type="` CounterSectionType `"`
`rows="` Integer `"` `columns="` Integer `"`
FrontLocation BackLocation
BackLocation
`supply="` Integer `"` `shadow="` Length `"` `/>`

CounterSectionType

- FrontSideOnly
- BackSideOnly
- TwoSided

FrontLocation

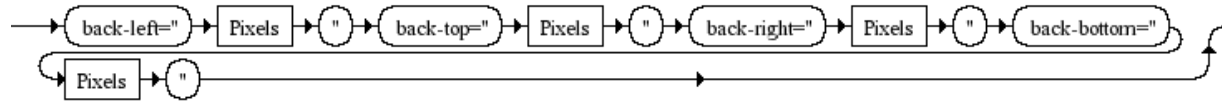`front-left="` Pixels `"` `front-top="` Pixels `"` `front-right="` Pixels `"` `front-bottom="`
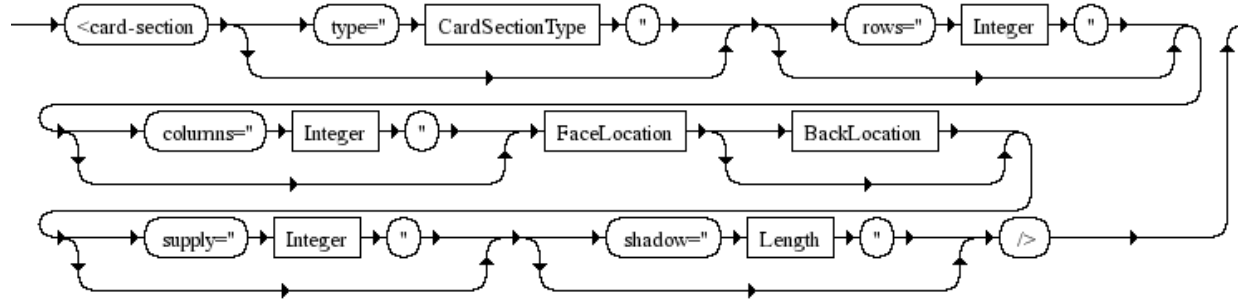Pixels `"`

The default value of "card-section type" is "CardFacesOnFront" or "CardFacesAndBackOnFront" depending on whether or not a back image is provided for the counter-sheet.
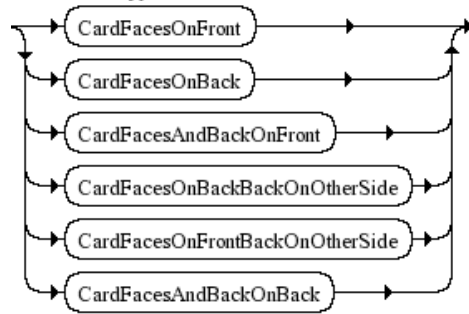
BackLocation

back-left=" → Pixels → " → back-top=" → Pixels → " → back-right=" → Pixels → " → back-bottom="

Pixels → "

CardSection

<card-section → type=" → CardSectionType → " → rows=" → Integer → "

columns=" → Integer → " → FaceLocation → BackLocation

supply=" → Integer → " → shadow=" → Length → " → />

CardSectionType

CardFacesOnFront

CardFacesOnBack

CardFacesAndBackOnFront

CardFacesOnBackBackOnOtherSide

CardFacesOnFrontBackOnOtherSide

CardFacesAndBackOnBack

FaceLocation

face-left=" → Pixels → " → face-top=" → Pixels → " → face-right=" → Pixels → " → face-bottom="

Pixels → "