

Untold Stories of The Secret World

Untold Stories of The Secret World is a MOD that allows you to write your own missions for The Secret World. This document contains instructions for writing missions using XML. Instructions for installing the mod are in `readme_untold.txt`.

If you like the idea of authoring a mission, don't let the use of XML scare you away. The schema has been designed to make writing missions easy. It feels more like writing a script than a complex XML document. Each XML tag will be discussed in detail, but the basic steps are as follows:

1. Create an XML file for your mission.
2. Add an entry to `MissionList.xml` for the XML file.
3. Add a tier to your mission.
4. Playtest the tier using the Local Mission launcher. As its name implies, this launcher is used for launching missions that are on your local computer, i.e. the XML file for your mission. The local launcher is available from a link on the main mission screen.
5. Repeat steps 3-5 until you have a complete mission.
6. Once you are satisfied with the results, publish the mission and make it available to all players. In the future, you will be able to do this directly in the mod. For now, [send Super Jenius a private message on the TSW forums](#), and I will post your mission.

TIPS

- XML files for three missions have been included with the mod. Take advantage of these files. Copy/paste tiers and adjust to your needs, or otherwise see how things work.
- Don't forget to include `<?xml version="1.0"?>` at the top of your XML file.
- XML files are case-sensitive, so be sure to respect the case of tags and attributes as described in this document.
- XML files require codes for some special characters. For example, `` indicates a carriage return, and `&` indicates an ampersand.
- Open the XML file using a web browser or run it through an XML validator to make sure the document is valid. XML is picky if you forget a closing tag or quote.

MissionList.xml

`MissionList.xml` contains a list of missions in XML files on your local computer. To be able to launch your mission, you need to add an entry to this file. This is what a mission tag looks like.

```
<mission title="My New Mission" id="MyMission" file="MyMission.xml" author="Super Jenius" />
```

Mission tag attributes:

- **title:** The title of your mission.

- **id:** The id of your mission, usually similar to your title, but it must be unique. It is used internally to track progress of the mission, so it must be different from other missions.
- **file:** The name of the XML file containing the mission.
- **author:** The author of the mission.

Mission XML File

The rest of the document describes elements of the mission XML document.

Mission Tag

The root element of the XML document is the **<mission>** tag. It is the top of the hierarchy. The entire mission will be contained within the starting **<mission>** and ending **</mission>** tags. Example:

```
<mission title="My Mission" debugMode="true" debugTier="3" >
```

Attributes:

- **title:** The title of the mission.
- **debugMode:** (Optional) When included and set to “true”, a debug window will appear on the left side of the TSW screen. This window allows you to move freely forward and backward through the mission while you develop it. It also may display information that is useful for certain tiers. Set to “false” or remove the tag to hide the debug window.
- **debugTier:** (Optional) Starts the mission at the specified tier, rather than the first tier. While writing a mission, you will be restarting often. Set this to go directly to the tier you are working on. Set to zero or remove the tag to start at the beginning of the mission

REMINDER

- Be sure to turn off the debug options before publishing your mission.

Tier Tags

Missions are made of up tiers. When a tier is completed, the mission proceeds to the next tier, and so on until the mission is complete. Tiers are small and consist of completing one action, such as targeting an NPC or listening to dialog. There are various types of tiers, which we will discuss in detail. There are some attributes all tiers share in common:

```
<tier type="tiertype" description="Tier description" faction="Dragon" gender="female" />
```

- **type:** The type of tier. Each type is discussed in detail below.
- **description:** The tier description will appear in the mission status window.

- **faction:** (Optional) You can vary the mission depending on which faction the player is in. When the tag is included, the tier will only run if the player is in the specified faction. Choices are “**Templar**”, “**Illuminati**”, and “**Dragon**”.
- **gender:** (Optional) As with the faction tag, you can vary the mission based on the player’s gender. When the tag is included, the tier will only run if the player is the specified gender. Choices are “**male**” and “**female**”.

Target Tier

The Target Tier completes when the specified target is selected (clicked on). This is useful before a Dialog Tier to make sure the player is near the desired NPC. It can also be used to detect when a specific named enemy is targeted. Please forgive the line break below. This is all one tag:

```
<tier type="target" description="Go see Jack Boone in Kingsmouth." targetName="Jack Boone"
isOffensiveTarget="false" />
```

- **targetName:** The name of the character to be targeted.
- **isOffensiveTarget:** When “false”, it indicates that the target is a defensive, friendly target, such as an NPC. When “true”, the target is an enemy, such as a monster.

TIP

- Not all NPCs are targetable. In that case, you can use a Location Tier to detect when the player is near a specific NPC before starting a Dialog Tier.

Location Tier

The Location Tier completes when the player comes within range of the specified coordinates.

```
<tier type="location" description="Go to the church." playField="3030" x="300" y="147" z="454"
distance="3" yDistance="1" />
```

- **playField:** The playfield (Kingsmouth, Savage Coast, etc.) ID that the desired coordinates are located in. [You can find a list of playfield IDs on the TSW forums.](#) The current playfield ID is also listed in the debug window when a Location Tier is active.
- **x, y, z:** The desired coordinates. NOTE: **x** and **z** are horizontal/vertical coordinates on the map. **y** is the height coordinate.
- **distance:** The distance to indicate the player is within range of the desired **x** and **z** coordinates.
- **yDistance:** The distance to indicate the player is within range of the desired **y** coordinate. **yDistance** will often be a lower number than **distance** to ensure the player is at the correct height.

TIP

- Turn on **debugMode** in the **<mission>** tag to see the player's current playfield and coordinates. Coordinates only appear when the player is currently in the designated playfield.

Kill Tier

The Kill Tier completes when the specified number of designated enemies have been killed.

```
<tier type="kill" description="Take out some zombies." targetName="Ravenous Horde,Returned Townie,Alerted Zombie" targetKills="12" />
```

- **targetName:** The name(s) of enemies/monsters that count as a kill. If multiple enemy types count as a kill, include the full name of the enemy, with each name separated by a comma.
- **targetKills:** The number of kills required to complete the tier.

Add Item Tier

This tier completes when the specified item is added to the player's inventory.

```
<tier type="additem" description="Buy a drink." itemName="Lengstereisen,Myrrh Ale,Mercurius,Bingo! Cola " autoUseItem="true" />
```

- **itemName:** The name of the item that should be acquired by the user. If one of several items can be acquired, then the full names of the items should be separated by commas. In the example above, buying any one of the four drinks will complete the tier.
- **autoUseItem:** (Optional) When "true", the item will automatically be used when it is acquired. In the example above, the acquired drink will automatically be used, which triggers a drink animation. If you would rather the player manually use the item, add a Use Item tier instead.

Use Item Tier

Tier completes when a specified item in the player's inventory is used.

```
<tier type="useitem" description="Drink the cola." itemName="Bingo! Cola" />
```

- **itemName:** The name of the item to be used. Only a single item can be specified, and it must exist in the player's inventory.

Browser Tier

This tier displays a specified URL in the in-game web browser. The tier completes when the browser is closed.

```
<tier type="browser" faction="Dragon" description="You have received a text message from headquarters." url="http://www.mbs-intl.com/tsw/scroll/scroll_dragon_text.png" browserTitle="Text Message" hideAddress="true" width="320" height="640" />
```

- **url:** The web address of the page to be displayed.
- **browserTitle:** (Optional) The title to be displayed in the web browser title bar. If omitted, the default “WEB BROWSER” appears.
- **hideAddress:** (Optional) When “true”, the browser address box and navigation buttons will be hidden.
- **width** and **height:** (Optional) The browser is resized to the specified width and height. This sets the size of the browser content, and the chrome around the content is in addition to that. For example, if you want to display an image that is 640x480, set the width to 640 and the height to 480. The actual browser window will be larger than that, but the image will display correctly.

TIPS

- The browser tier is a powerful feature for displaying not only web pages, but also images and video. It is especially valuable for creating investigation missions.
- If you want the player to input something like a password or an answer to a question, process the input then provide info on the next tier within the web page. This tier cannot verify a response from the player. It can only display content.
- Flash content is not supported by the in-game browser. However, many HTML5 features are supported, including animations, audio, and video.
- While HTML5 audio is supported, MP3 is not. The Ogg Vorbis (.ogg) format is recommended for audio. There are web sites that make it easy to convert an MP3 file to Ogg format. I believe WAV is also supported, although WAV files are much larger than Ogg.
- HTML5 videos on YouTube work, although they are limited to 360p, so the quality is relatively low.
- When setting the size of the browser window, keep in mind that some players may be running at a lower resolution. Somewhere around 960x640 is probably a safe maximum size for most players.
- If the content needs to scroll, it’s best not to change the size from the default. Otherwise, the scroll bar might show up in an odd position.

Audio Tier

The Audio Tier is used to play audio in the background during a mission. There are similar tags in Dialog and Cinematic tiers, which can be used to play audio and voiceover during those tiers. The Audio Tier will often be used to preload audio that will be used later in dialog and cinematics, ensuring the timing is consistent.

IMPORTANT: MP3 files are not supported, and may crash the game if tried. Ogg Vorbis files are recommended. WAV is also supported, but result in much larger files. Online sites are available that easily translate MP3, WAV, and other formats to OGG.

```
<tier type="audio" url="http://www.mysite.net/mydialog.ogg" preload="true" />
```

- **url:** Audio files are downloaded from the web. This is the url to the file.
- **preload:** (Optional) When “true”, the file is preloaded instead of being played immediately. This is useful when preloading voiceover for a dialog/cinematic tier where timing is critical. If you don’t preload, the audio file will take a few seconds to download before starting, and timing will be unpredictable. For example, you can insert an audio preload tier while the player is on their way to meet an NPC that will start the dialog. Then the dialog/cinematic tier will actually play the file.

```
<tier type="audio" url="http://www.mysite.net/siren_song.ogg" volume="50" loop="true" />
```

- **volume:** (Optional) The volume to play the file (0 -100). If omitted, the audio is played at full volume (100).
- **loop:** (Optional) If “true”, the audio will loop continuously.

```
<tier type="audio" stop="true" />
```

- **stop:** Insert the tier above to stop audio that started playing in a previous audio tier. Otherwise, sound will continue to play until the audio file is complete (unless looping) or the mission ends.

Dialog Tier

The Dialog Tier is used to display multiple lines of dialog as subtitles. The tier is complete when all lines of dialog in the tier have been displayed. The main <tier> tag contains multiple child <dialog> tags for the lines of dialog. Here is an example tier:

```
<tier type="dialog" description="Listen as Jack Boone talks about the dialog system." speed="2.5" >
```

```
<dialog type="dialog" gender="male" line="{{m_Player.m_Name}}... How's the  
{{m_Player.m_Faction}} treatin' ya, brother?" />
```

```
<dialog type="dialog" gender="female" line="{{m_Player.m_Name}}... How's the  
{{m_Player.m_Faction}} treatin' ya, darlin'?" />
```

```
<dialog type="dialog" line="You need a pause in the conversation, you just fade out for a bit..." duration="6" />
```

```
<dialog type="fadeout" duration="1" />
```

```
<dialog type="dialog" line="... then fade back in... " />
```

```
<dialog type="fadein" duration="1" />
```

```
<dialog type="dialog" line="... and continue your dialog." />
```

```
</tier>
```

Let's first look at the attributes in the main <tier> tag.

- **speed:** (Optional) allows you to control the pace of the conversation. There is a built-in equation that controls how long dialog is displayed without having to set the duration for each line. The default is "3", which means roughly 3 words per second plus some padding. If you want to slow down the dialog a bit, set the speed to a lower number, such as 2 or 2.5. To speed things up, go with a slightly larger number.

Within the <tier> tag are multiple <dialog> tags, each with a type specified. I'll describe each one.

"dialog" Type

- **line:** The line of dialog to appear in the subtitle.
- **duration:** (Optional) If you want to override the calculated duration for a specific line, you can set it with this attribute. The duration is specified in seconds.
- **gender:** (Optional) Similar to <tier> tags, you can vary the conversation based on the player's gender. In the example above, the first line only appears if the player is male, and the second line appears if the player is female.
- **faction:** (Optional) Similar to <tier> tags, you can vary the conversation based on the player's faction, "Templar", "Illuminati", or "Dragon".
- There are some special expressions you can include within the line of dialog. These expressions must be surrounded with double curly braces:
 - **{{m_Player.m_Name}}**: The name of the player.
 - **{{m_Player.m_Faction}}**: The faction of the player.
 - **{{m_Player.m_Gender}}**: The gender of the player.

"fadeout"/"fadein" Types

- **type:** "fadeout" means the screen should fade to black. "fadein" means the screen should fade back in from black.
 - **duration:** Indicates how long the fade should take.
-

TIPS

- When setting the dialog speed, keep in mind that all players do not read at the same rate. I try to imagine what the speed of the dialog would be if it were voice acted. This usually provides adequate time for most people to read without irritating fast readers. In general, the default speed should be ok, but complex conversations or lines with “big” words may need to be slowed down a bit.
- Fadeout/in lines are good to denote that some time has passed or some “off-screen” event has occurred between two lines of dialog.

“audio” Type

```
<dialog type="audio" url=" http://www.mysite.net/mydialog.ogg" volume="20"/>
```

See the Audio Tier section for parameter descriptions. As mentioned in that section, if using voiceover, it will be helpful to preload the audio file in an audio tier first, then play the same file in the dialog tier.

Cinematic Tier

The Cinematic Tier is basically a Dialog Tier with the addition of camera work. A cinematic produced by FunCom has motion capture, voice acting, camera positioning, and pretty much anything else FunCom wants to do with the engine. By comparison, we have camera positioning and subtitles. Previous versions of the TSW API also allowed the placement and rotation of the player and NPCs, but those functions have since been removed (which I grumble about). We do now at least still have the capability to set which direction the player faces. Overall, we cannot achieve the same results as FunCom. That said, I do believe the camera work adds quite a bit to the experience compared to subtitles only. Getting the camera work right can be time consuming, so I consider cinematics as an optional but worthwhile improvement. Play The Scroll mission and decide for yourself if cinematics are worth the additional effort.

The cinematic <tier> tag is the same as the dialog <tier> tag, with the addition of one attribute:

```
<tier type="cinematic" description="Listen to the Oxford archeologists." playField="3090" >
```

- **playField:** Set this to the playfield where the cinematic will occur. Camera work will only occur if the player is in the specified playfield. Otherwise, the tier will function like a Dialog Tier and only subtitles will appear. This is used if you have previously played a cinematic, then want to replay the dialog (using the Back (<) button) without going back to the original location of the cinematic.

Within the <tier> tag are multiple <cinematic> tags, each with a type specifying the action to take. Types “dialog”, “fadeout”, “fadein”, and “audio” are identical to the Dialog Tier, so I won’t repeat them here. Here are other types described in detail:

“camera” Type

This type sets a camera position and target coordinates that the camera will focus on. This can be used to set the camera to a fixed position. It can also be used to set a camera path and how long it takes to get from point A to point B. Camera paths are usually set in pairs, one to set the start of the camera path and another to set the end of the path.

```
<cinematic type="camera" duration="0" posX="714" posY="52.5" posZ="858.8" targetX="715.6" targetY="52.8" targetZ="858.3" />
```

```
<cinematic type="camera" duration="30000" posX="713" posY="52.5" posZ="859.8" targetX="715.6" targetY="52.8" targetZ="858.3" ease="Regular.easeIn" />
```

- **duration:** This is the amount of time (in milliseconds) designated for the camera to move from its current position to the specified position. Set this to zero to immediately move the camera to the position, i.e. to set the camera to a fixed position or to set the start of a camera path. Add a second tag with a duration to define the end of the camera path and how long it will take to get there. Timing can be confusing and is explained in more detail below. NOTE: Camera duration is specified in milliseconds, while dialog line duration is specified in seconds. I apologize for any confusion this may cause.
- **posX, posY, posZ:** The coordinates where the camera is placed.
- **targetX, targetY, targetZ:** The coordinates the camera will focus on. Set **targetX** to “-99” to focus on the player (**targetY** and **targetZ** will be ignored).
- **ease:** (Optional) An “ease” function changes how quickly the camera moves in and out of a camera path, which can provide a more natural movement. Refer to [this site](#) for a list of available ease functions and examples of their movement.

“facetarget” Type

Set the player rotation to face the currently selected defensive target/NPC.

```
<cinematic type="facetarget"/>
```

“facenpc” Type

Set the player rotation to face the named NPC. This is useful for having the player face named NPCs that are not targetable, or that otherwise would not be the currently selected target.

```
<cinematic type="facenpc" name="Richard Sonnac"/>
```

- **name:** The name of the NPC to face.

“facecoords” Type

Set the player rotation to face the specified coordinates.

`<cinematic type="facecoords" x="335.4" z="301.47"/>`

- **x, z:** The coordinates the player should face. You can't make the player look up or down, so there is no **y** coordinate.

Cinematic Timing

The timing of cinematics is difficult to explain. In short, the duration of a camera tag controls how long it takes for a camera to move from one position to another. For example, if you are zooming in on a character, the duration sets how quickly you want to zoom in. However, the timing between multiple camera paths is controlled by the duration of dialog lines. This was done to ensure camera paths are synchronized with the dialog. As such, new camera positions and paths are set directly before a line of dialog. Like I said, hard to explain. It's easier to see in action. Here is a short example. The lines in red explain what the next line is doing and are not included in the XML file:

`<tier type="cinematic" description="Listen to the Oxford archeologists." playField="3090" >`

Fade to black. Take 2 seconds for the transition.

`<cinematic type="fadeout" duration="2" />`

Add a short pause (a dialog tag with an empty line), which keeps the screen black for another half-second before proceeding.

`<cinematic type="dialog" line="" duration="0.5" />`

Set the player to face Montgomery.

`<cinematic type="facenpc" name="Montgomery de la Roche"/>`

Set the start of the camera path focusing on Montgomery. Since duration is zero, the camera is set to this position immediately.

`<cinematic type="camera" duration="0" posX="714" posY="52.5" posZ="858.8" targetX="715.6" targetY="52.8" targetZ="858.3" />`

Set the end position of the camera path. Slowly zoom in on Montgomery over up to 30 seconds. Camera movement begins immediately when this tag is processed.

`<cinematic type="camera" duration="30000" posX="713" posY="52.5" posZ="859.8" targetX="715.6" targetY="52.8" targetZ="858.3" />`

Fade in from black over 5 seconds. The camera is moving during the fade in.

`<cinematic type="fadein" duration="5" />`

When fade in is complete, display Montgomery dialog. The camera is still moving towards the end of the camera path, i.e. zooming in on Montgomery.

<cinematic type="dialog" line="Montgomery: I'm telling you, Arun. This mystery goes deeper. We must learn more!" />

As soon as Montgomery dialog is complete, start a new camera path. This will take less than 30 seconds, and the new path will start before the previous path is complete. This is ok and in fact usually desirable. More on this in the TIPS section below.

Set the start of the camera path and focus on Arun.

<cinematic type="camera" duration="0" posX="714" posY="52.5" posZ="858.8" targetX="715.8" targetY="52" targetZ="861.2" />

Set the end position of the camera path. Slowly zoom in on Arun over up to 30 seconds.

<cinematic type="camera" duration="30000" posX="713" posY="52.5" posZ="857.8" targetX="715.8" targetY="52" targetZ="861.2" />

Display Arun dialog. The camera is still moving towards the end of the camera path, i.e. zooming in on Arun.

<cinematic type="dialog" line="Arun: There is no mystery! It is a blank piece of papyrus!";We have more pressing matters to attend to." />

When Arun dialog is complete, reposition camera and focus on the player. Player position can vary, so targetX="-99" tells the camera to look at the player's current position.

<cinematic type="camera" duration="0" posX="716" posY="52.5" posZ="859.7" targetX="-99" targetY="0" targetZ="0" />

Change the camera position over 30 seconds while continuing to focus on the player.

<cinematic type="camera" duration="30000" posX="723.3" posY="56.2" posZ="857.3" targetX="-99" targetY="0" targetZ="0" />

Display next line of dialog. Camera continues to move while the line is displayed.

<cinematic type="dialog" line="Perhaps you can settle the matter for us..." />

After the last line of dialog is displayed, fade out over 3 seconds to indicate the end of the cinematic.

<cinematic type="fadeout" duration="3" />

Don't worry about fading back in. It will happen automatically when the cinematic tier is complete.

</tier>

TIPS

- We talked a lot about camera movement paths, but fixed camera positions are ok too. You can simply point the camera in the desired direction, set the duration to zero, and leave it alone. Since there is no movement, only one camera tag is needed.
- As in the example above, the duration of a camera path should often be set longer than the duration. Don't try to time the durations of the camera and dialog exactly. If the camera path ends before the dialog, the camera will stop moving and remain in the last position, which may feel a little abrupt to the player. **UPDATE:** When using an ease function, it is more important to get the duration more exact.
- A camera position or path can span multiple lines of dialog. It doesn't need to change with every line of dialog. The camera will remain at the current position or on the same path until a new position or path is set.
- In The Scroll mission, I demonstrate a lot of camera movements, which means some are overdone. Slow, subtle camera movements are a better rule most of the time.
- If you want to set a camera in a fixed position focused at a character, adding very slow camera movement is a nice touch. Set the end of the path a short distance away and set the duration high (hundreds of thousands of milliseconds).
- Use the player rotation functions (facetarget, facenpc, facecoords) sparingly. They cause a slight pause in the game client, and using them too often could affect game stability. Limit it to once or twice per cinematic, and you should be ok.
- As an alternative to in-game cinematics/dialog, you can choose to create web pages and advance the mission/story using the Browser Tier. This may be preferable for authors that have artistic and/or Photoshop skills. Both approaches are valid, so use the one that suits you.