

Une erreur est survenue lors du chargement de la version complète de ce site. Veuillez vider le cache de votre navigateur et rafraîchir cette page pour corriger cette erreur.

Séquences d'échappement de l'interface utilisateur

iThorgrim

Séquences d'échappement de l'interface utilisateur

L'Intemporel

Important : Ce tutoriel a été rédigé par L'Intemporel pour le site wow-emu.fr.

Je le recopie ici à l'identique, Il se peut que certaines informations soient obsolètes à ce jour. Bonne lecture !

Sommaire :

- Qu'est ce qu'une séquence d'échappement
- Colorer un texte ou une partie de celui-ci
- Ajouter une images/textures
- Les liens/hyperliens
 - Les différents 'LinkType' avec leurs 'LinkData'
 - Comprendre leurs fonctionnement
- Autres
- Conclusion

Qu'est ce qu'une séquence d'échappement

On entend par séquence d'échappement (de l'anglais escape sequence) une séquence de quelques octets et/ou de caractères présente dans un flot de texte jouant un rôle particulier à l'endroit où elle est présente. Elle est introduite par un code prévu à cet effet.

Exemple : Sur *World of Warcraft* une des possibilité des séquences d'échappement est d'introduire un bout de code à même un texte qui lui permet d'avoir une couleur différente de celle de base.

Colorer un texte ou une partie de celui-ci

La séquence d'échappement pour mettre un texte ou une partie d'un texte en couleur est la suivante :

```
|cAARRGGBB texte|r
```

Ici **|c** sert à déclarer le début de la séquence d'échappement et **|r** sert à fermer la séquence.

Il nous reste alors 4 différentes variables prenant des valeurs de 0 à 255 en hexadécimal, de 00 à FF :

- AA - Canal Alpha, 00:transparent ; FF:opaque
- RR - Canal Rouge
- GG - Canal Vert
- BB - Canal Bleu

Exemple :

```
|cFFFF0000Ceci est rouge|r
```

Ajouter une images/textures

Avec les séquences d'échappement il est aussi possible d'ajouter une image dans un texte, voici la structure de cette séquence :

```
|TTexturePath:size1:size2:xoffset:yoffset|t
```

Comme pour les couleurs on retrouve une partie qui ouvre et clôture cette séquence, ici c'est **|T** et **|t** respectivement.

Faites bien attention entre majuscules et minuscules, c'est important !

Ici il y a plusieurs choses à savoir. Tout d'abord **size2** est facultatif (on va voir pourquoi après). Ensuite **xoffset** et **yoffset** sont aussi facultatif ils servent uniquement à déplacer la texture de la où elle devrait être.

TexturePath est le chemin vers la texture, voici par exemple un chemin vers une texture :

Interface\Icons\INV_Misc_Coin_01

Il ne faut pas mettre d'extension à la fin, et penser à doubler les anti-slash.

Ensuite il existe plusieurs cas différents pour la taille de la texture : **size1** et **size2**, pour cela on va ajouter d'autres variables qu'on va appeler **TextH** qui est basé sur la taille de la hauteur de la police utilisé, puis Hauteur et Largeur qui représente respectivement la hauteur et la largeur de la texture qui va être affiché.

- Si size1 == 0 et size2 ne prend pas de valeur, **alors Hauteur = Largeur = TextH**
- Si size1 > 0 et size2 ne prend pas de valeur, **alors Hauteur = Largeur = size1**
- Si size1 == 0 et size2 == 0, **alors Hauteur = Largeur = TextH**
- Si size1 > 0 et size2 == 0, **alors Hauteur = size1 et Largeur = TextH**
- Si size1 == 0 et size2 > 0, **alors Hauteur = TextH et Largeur = size2 * TextH**
- Si size1 > 0 et size2 > 0, **alors Hauteur = size2 et Largeur = size1**

Note :

Lorsque vous voulez utiliser une icône carrée dans un texte (spell/item icône) utilisez uniquement : `|TTexturePath:0|t`

Les liens/hyperliens

Il existe différent type de lien pouvant être utilisé par les séquences d'échappement, un des exemples de liens est par exemple l'affichage d'une fenêtre d'informations concernant un item lorsque l'on survole son nom.

Mais il existe d'autres liens qui permettent d'afficher des quêtes, etc...

Ces séquences d'échappement sont vraiment très utiles, néanmoins elles restent difficiles à utiliser car chaque options prend en compte plein de paramètres différents.

Voici la structure générale de ces liens :

```
|HLinkType:LinkData|hLinkText|h
```

C'est le 'LinkType' est ce qui définira quel type de liens cela va être.

Les nombreux ":" qui se suivront à la fin du 'LinkData' (possibilité d'y en avoir qu'un seul) sont pour définir à 'nil', les quelques variables inutilisées de certaines séquences, leurs suppressions peuvent être réalisés sans problèmes, néanmoins c'est un détail que je tiens en considération.

Les différents 'LinkType' avec leurs 'LinkData'

'LinkData' si possible de nommer leur appartenances et uniquement les plus utiles

- achievement
 - achievementID
 - playerGUID
- enchant
 - enchantID
- glyph
 - socket
 - glyphID

- instancelock
 - playerGUID
 - instanceID
 - difficulty
 - bossesDefeated
- item
 - itemID
 - enchant (optionnel)
 - gem1, gem2, gem3, gem4 (optionnel)
 - suffixID (optionnel)
 - uniqueID (optionnel)
 - level
 - upgradeID
 - instanceDifficultyID
 - nomBonusIDx
 - bonusID1, bonusID2, ..., bonusIDx
- journal
 - contentType
 - contentID
 - difficulty
- levelup
 - level
 - type
- player
 - (eg. à venir)
- playerGM
 - (eg. variation de 'player', uniquement pour le dialogue pour un GM)
- quest
 - questID
 - levelRecommended (optionnel)
- spell
 - spellID
- talent
 - talentID
 - rank (optionnel)
- trade
 - spellID
 - skill
 - maxSkill
 - playerGUID
- channel
 - command
 - name

Il existe énormément de 'LinkType' certains plus utiles que d'autre, je vais détailler les plus complexe et vous laisserez réfléchir aux autres.

Pour la définition des 'LinkData', sachez que les variables doivent être séparées par deux points ":", eg. :

```
|Htrade:spellID:skill:maxSkill:playerGUID:|h
```

Achievement

Achievement

Ici comme vous pouvez le comprendre, c'est pour les Hauts-Faits, la première variable 'achievementID' c'est pour l'ID de l'haut-fait et la seconde variable est le GUID du joueur, codé en hexadécimal.

Il existe une fonction en lua qui retourne le GUID d'un joueur directement en hexadécimal, néanmoins ce GUID sera accompagné du préfix "0x", chose qu'il faut enlever. Cette fonction est UnitGUID("player")

```
|Hachievement:2336:0600000000279E425:::|h[Insane in the Membrane]|h
```

Ici cette séquence vous affichera le Haut-Fait portant l'ID 2336 avec toutes les informations de complétion du joueur portant le GUID (hex) 0600000000279E425.

Enchant

Ici cette séquence va vous permettre d'afficher un objet comme si vous le linkez depuis la recette de fabrication du métier qui permet de le créer. La variable ici présente pointe donc vers l'ID d'un spell.

```
|Henchant:59387|h[Acte de Propriété]|h
```

Glyph

Cette séquence va vous permettre d'afficher les informations relatives à une glyphe, néanmoins je ne sais pas trop à quoi représente la variable 'socket', ici 23 marche bien pour afficher la glyphe portant l'ID 460

```
|Hglyph:23:460|h[Glyph of Fortitude]|h
```

InstanceLock

Ici c'est un peu plus compliqué déjà, le playerGUID est comme tout à l'heure, c'est le GUID du joueur en valeur hexadécimal, l'instanceID est l'ID de l'instance en question (eg. 531 pour le Temple d'Ahn'Qiraj), la difficulté est :

0. normal
1. héroïque 5
2. héroïque raid

Quand à 'bossesDefeated' il faut déjà plus réfléchir, en effet dans cette variable il faudra mettre un masque de bit, qui indique les boss vaincus et ceux qu'il ne le sont pas, pour trouver cette valeur il suffit tout d'abord de savoir l'ordre des boss dans le donjon, ainsi que leur nombre.

Ici on va prendre au hasard un donjon en admettant qu'il y a 5 boss, le masque de bit de cette variable sans avoir tué aucun boss sera le suivant :

```
00000, soit - 0
```

Soit en valeur numérique : 0.

Si le joueur arrive à tuer le premier et le troisième boss, son masque de bit sera alors :

```
00101, soit - 101
```

Ce qui donne 5 en valeur numérique.

Maintenant que vous avez compris le fonctionnement voici les différents masque de bit que l'on peut avoir pour ce donjon qui compte 5 boss :

```
bitmask - boss tué
-----
00000 - aucun
```

```
00001 - 1
00010 - 2
00011 - jusqu'à 2
00100 - 3
00101 - 1,3
00110 - 2,3
00111 - jusqu'à 3
01000 - 4
01001 - 1,4
01010 - 2,4
01011 - 1,2,4
01100 - 3,4
01101 - 1,3,4
01110 - 2,3,4
01111 - jusqu'à 4
10000 - 5
10001 - 1,5
10010 - 2,5
10011 - 1,2,5
10100 - 3,5
10101 - 1,3,5
10110 - 2,3,5
10111 - 1,2,3,5
11000 - 4,5
11001 - 1,4,5
11010 - 2,4,5
11011 - 1,2,4,5
11100 - 3,4,5
11101 - 1,3,4,5
11110 - 2,3,4,5
11111 - jusqu'à 5
-----
```

Ainsi voici un exemple :

```
|Hinstancelock:01000000003E8E2E7:531:0:15|h[Ahn'Qiraj Temple]|h
```

Cela signifie que le joueur avec le GUID 01000000003E8E2E7, dans l'instance 531 en mode normal (0) à tué les 3 premier boss (bitmask = 15 - 111), et donc le lien affichera les informations suivantes dans une frame.

Channel

Cette séquence va vous permettre d'afficher un channel (canal de discutssion) qui va vous permettre lors du clique de le rejoindre aussitôt ! Par exemple en jeu vous pouvez très bien avoir le canal de discussion [1. Général - {zone dans laquelle vous vous trouvez}] et lors du clique, vous aurez directement la possibilité d'écrire dans ce canal là.

```
|Hchannel:channel:{id}:{[nom du canal]}|h
```

Ce code-ci vous affichera donc dans le chat (là où vous l'avez placé) : {[nom du canal]}, et lorsque vous cliquerez dessus, vous aurez directement la possibilité d'écrire dans le canal portant l'id : {id}, pour le canal Général c'est donc :

```
|Hchannel:channel:1:[1. Général - {zone}]|h
```

Ce qui se passe exactement lors du clique c'est que le client fait a votre place cette suite de "commande" :

```
{ouvre la barre de chat} -> effectue : /{id} -> (valide)
```

Autres

Il existe d'autre séquence d'échappement qui peuvent vous être utile, certaines très compliqué d'utilisation et d'autre non, c'est pourquoi je vous passe seulement la plus utile.

Si vous voulez écrire sur une nouvelle ligne, il vous faut donc introduire la séquence : `\n` (il faut néanmoins que le Widget supporte l'affichage car certains ne vont pas prendre le saut de ligne et donc ne pas se 'resizer')

Conclusion

Les séquences d'échappement sont quelques choses de très utiles si vous voulez formater un temps soit peu vos texte.

J'espère que ça vous aura plus, bonne fin de journée,
L'Intemporel
