Université de Genève
Faculté de Psychologie et de Sciences de l'Education
Unité "Technologies de Formation et Apprentissage"

TecfaMOO

Manuel

Daniel K. Schneider & friends [1]

schneide@divsun.unige.ch

(Daniel et Kaspar sur TecfaMOO.unige.ch 7777)

Janvier 1995

(brouillon!, produced with Webmaker)

1. Le TecfaMoo est un nouveau projet qui se précise au niveau international. Il a pu voir le jour sous cette forme seulement grâce à notre team de wizards, dans l'ordre chronologique: ***xymox, irradiate, gustavo et janus***. Ces personnes venues du "cyberspace" international nous ont d'abord construit rapidement un "coeur" de haut niveau et continuent à nous fournir les outils essentiels dont on a besoin pour amorcer notre projet d'espace virtuel pour l'éducation et la recherche. Je tiens à les remercier vivement ici.

1. Introduction

Note for english readers: You may not understand much, but glancing at the list of commands may be useful (once available).

Un “MOO” est un serveur de réalité virtuelle textuelle (RVT). Il s’agit d’une sorte de combinaison entre les forums du Télétexte (ou Compuserve) et les jeux d’aventures textuelles en y rajoutant toute la gamme de logiciels asynchrones de télécommunication dont on a habitue. Cette technologie est à nos yeux (et sur le plan fonctionnel) le précurseur le plus important du futur “cyberspace”.

NOTE: Sur tous les MOO il existe un certain nombre de commandes standards. Las plupart des commandes décrites ci-dessus fonctionner dans le TecfaMOO, les autres vont y être rajoutées. La majorité des commandes sont non-standards et peuvent ou ne peuvent pas marcher sur d’autres MOO. Toutefois, on sépare ici les commandes standards partout des extensions.

1-1 Clients MOO (Moo-Clients)

Pour travailler avec MOO, il est très fortement conseillé d’utiliser un client spécialisé. Il est difficile de travailler avec une connexion telnet.

1-1.1 Sous Emacs: le client “rmoo”

M-x rmoo
: Vous propose le choix d’un Moo et vous interroge éventuellement sur votre nom + mot de passe. La liste des MOOs disponibles et stockée dans le fichier ~/.rmoo_worlds

c-c c-q **Quitter une séance MOO**

M-p **Répéter (ou plutôt reafficher) la commande précédente**

1-1.2 Muddweller sur MAC

- Utilisez les menus pour vous connecter
- Tapez vos commandes dans la toute petite fenêtre en bas du client.

1-1.3 (tf) Tinyfugue sous Unix

- Tapez “tf <adresse> <no porte>”
- la commande Unix “tecfamoo” lance tf pour le TecfaMOO.

/quit **pour déconnecter**

/visual on **sépare la ligne input du reste**

/help **donne la liste des commandes**

1-2 Connexion/Déconnexion sur un MOO

connect **<user-name> <mot-de-passe)**
: Pour se connecter

Notez que vous pouvez configurer la plupart des client pour faire une connexion automatique

@quit **pour quitter**

@password <old password> <new password>
: Change le mot de passe

1-3 Aide (Help)

help **Affiche une liste de “topics”**

Arguments:

< topic> **Affiche une introduction à un “topic” / liste de commandes disponibles**

Exemple:

(1) help movement

Affiche une aide sur la navigation dans l’espace “MOO”

<commande> **Affiche l’aide pour une commande**

Exemple:

(1) help page

Affiche une aide pour la commande “page”

(2) help #<no objet>

Affiche l’aide pour un objet (si disponible)

(3)help #<no objet>:<verbe>

Affiche l’aide pour un verbe attaché à un objet

1-4 Configuration de votre fenêtre terminal

@wrap ont to make the MOO wrap the text you read

@linelength XX to set the length of your lines, in letters, to XX

@pagelength XX to set the length of your page, in lines, to XX

@more rest/flush to read/flush suspended output

2. Communication

2-1 Communication de base

Dans les exemples qui suivent vous êtes “John”

say **Dire quelque chose à tout le monde dans une salle**

“ **(abréviation)**

Exemple:

(1) Hello, there.

Vous allez voir: You say, "Hello, there."

(2) "Hello, there

emote **Expression “non-verbales”**

: **(abréviation)**

Exemple:

(1) emote smiles.

Vous allez voir: John smiles.

(2) :smiles

(la même chose)

whisper **communication privée**

Exemple

(1) whisper “Hello, Jane darling” to Jane

Visible seulement pour Jane (si elle se trouve dans la même salle). Vous allez voir: You whisper, "Hello, Jane darling" to Jane.

@who **Afficher toutes les personnes connectées au MOO**

@who lister les gens

@who <nom>... donne de l’information sur une personne

page **Communication à distance**

Exemple:

(1) page Jane Do you have some free time?

2-2 Extensions du TecfaMOO (et d’autres)

- **S’adresser publiquement à une seule personne**

‘ **(abréviation, backquote!)**

Exemple:

(1) -Jane I agree with you.

Vous allez voir: 'John [to Jane]: I agree with you.

(2) - “... I agree again

Avec “-” sans arguments on s’adresse de nouveau à la même personne

think **“penser” (Visible pour tout le monde dans une même salle)**

Exemple:

(1) think I think therefore I am.

Vous allez voir: John . o O (I think therefore I am)

+ **“emote” à distance**

Exemple:

(1) +guest waves

Peut donner: Jane waves to you (si vous êtes “guest”).

= **CB (utilisation d’un canal radio sur TECFAMOO et EON)**

Il faut avoir la feature #122 (@addfeature #122), faites help #122 pour avoir toutes les commandes.
Example:
(1)=Je suis perdu, HELP !
Tout le monde branché sur le même canal peut voir ce message, à utiliser avec modération

2-3 Lien Inter-MOO (sur TECFA, DUM, Eon, Gna-Lab, etc.!)

Il existe un lien entre un certain nombre de MOO éducatifs: Pour utiliser ces fonctionalités il faut: @addfeature #310.

help #310 ..... pour avoir toutes les commandes

look #307 ..... pour avoir la liste des serveurs MOO connectées.

rwho **Lister les personnes connectes à un autre serveur**
Exemple:
(1) rwho kaspar@eon
Voir si Kaspar est connecté sur Eon
(2) rwho @eon
Lister toutes les personnes connectées sur Eon

rpage **Envoyer un message à une personne sur un autre MOO**
Exemple:
(1) rpage guest@dum "Hi, how are you?"

rep **Répondre au dernier rpage reçu**

2-4 Communication asynchrone/Systèmes d'information

La plupart des MOO ont des "News", "Poster", "e-mail", etc. internes. Examiner ces objets pour savoir comment les utiliser. Certains MOO ont des interfaces simples à "gopher" et "www".

@subscribe (Lister les Newsgroupes disponibles)

@mail **Lister le contenu d'une collection de courier**
Exemples:
(1) @mail
Liste la collection de votre courier personnel
*(2) @mail on *new*
Liste les messages dans la collection *new

@read 1 on <newsgroup> (Afficher le message 1 d'un newsgroup)
*(1) @read 1 on *news*

@rn **Lister les collections de News et de courrier ayant reçu de nouveaux messages**

@nn (*) **Lire tous les nouveaux messages dans toutes les collections abonnées**

@send **Envoyer un (moo)email (à compléter!)**
(1) @send daniel
"message test
"Comment ca va ?
send
Envoye un message à Daniel avec sujet "message test". Notez que vous devez "parler" au programme mail et terminer avec la commande "send".

@qsend **quicksend mail**
(1) qsend lucy je n'ai pas pu rester ... a la prochaine
Envoyer un mail à Lucy ...
(2) qsend "lucy" "nina" subj="oops" Je n'ai pas pu vous attendre ---- X.Y.
Envoyer un message à lucy ET nina avec une ligne de sujet. Notez l'utilisation des quotes. Pour plus de détail, faire "help me:@qsend".

3. La Navigation

go **marcher dans une direction**

En règle générale il faut lire la description d'une salle. Elle indique les sorties (par ex "north", "south", "out"). En tapant ces directions ou "go <direction>" on peut se déplacer.

Les commandes suivantes n'existent que dans certains MOO:

@addroom se souvenir d'une "chambre"

(1) @addroom atrium #101

Tapez "@examine here" pour afficher le numéro d'une salle

@rooms **Afficher la liste des chambres mémorisées**

Extensions

map **afficher une "carte" de la région ou du MOO (pas tjrs disponible)**

Notez que sur la plupart des MOO il existe souvent des cartes (maps) qu'il faut lire avec la commande "read map"

@go **téléportage vers un objet**

@join **joindre une personne à un endroit**

(1) @join Daniel

Vous joignez Daniel à l'endroit où il est

Notez qu'il est poli de "frapper" avec knock ou de faire un "page" avant de joindre une personne qui se trouve visiblement pas dans un endroit public.

@knock **frapper "à la porte" d'un "room"**

(2)@kock daniel

Cette commande sert à demander si on peut @join une personne à un endroit donné.

@invite **inviter une personne à se joindre,**

Utilisé souvent pour répondre à un "@knock", "help @knock" pour plus de détails
Exemple:

(1) @invite

Invite la dernière personne qui a frappée

(2) @invite Jean

Invite Jean à se joindre

4. Les objets du MOO (Moo objects)

4-1 Introduction

Tout objet, personnage etc. sur un MOO est également un objet au sens informatique. Il porte un numéro (cf. la commande “examine”) qui l’identifie. Dans une même salle tous les objets sont également identifiables et manipulables par leur nom ou par un nom d’alias. Lorsque deux objets portent un même nom il faut utiliser un de ces autres noms (si disponible) ou son numéro d’objet pour le manipuler.

4-2 Examen et manipulation simple d’objets

look **Examiner un objet (afficher une description)**
Exemples:
(1) look
Afficher la description de la salles
(2) look board
Regarder l’objet “board”

examine **Examiner un objet en détail (y compris quelques infos techniques)**
Exemple:
(1) examine board

inv **Lister l’inventaire des objets que vous portez**

get **Prendre un objet (take=synomyme)**
Exemples:
(1) get boite
Prendre la boite qui se trouve “par terre”
(2) get boite from box
Prendre la boite qui se trouve dans la box

drop<objet> Laisser tomber un objet par terre

put **Mettre un objet quelque part**
Exemple:
(1) put boite in box

give **Donner un objet**
Exemples:
(1) give Pet to Daniel
Donner l’objet “pet” à l’utilisateur Daniel

5. Les personnages (Players)

5-1 Description du personnage

Vous pouvez modifier votre personnage avec les commandes suivantes:

@describe me as " .... " (description)

(1) @describe me as "Innocent, mais capable d'apprendre"

@gender Définition du sexe

(1) @gender male
(2) @gender female

@addalias Ajouter un alias au nom du personnage

@rename

Exemples:

(1) @rename me to Chien_mechant

Se renommer "chien_mechant"

(2) @rename me to Fleur, Joelle

Se renommer comme "Fleur" avec un alias "Joelle".

Pour regarder une personne, utilisez la commande "look", dans certains moo il existent les commandes suivantes (pafois sans le signe "@":

finger donne de l'information sur la provénance et la connexion d'une personne (ou @finger, whois, @whois)

Exemple:

(1) finger Julie

Affiche des informations sur Julie (et son état de connexion)

Pour donner une description à votre personnage, faires '@notedit me.whois_msg'. Cf. la section 6-3 "L'édition de texte" [p. 11] pour savoir comment éditer un texte.

Sur certains MOO il existe en plus (ou alternativement) une commande @research ou @interests.

@mood donne un "mood" qui s'affiche lorsqu'on utilise la commande "who"

(1) @mood "occupe"

... montre en principe que vous êtes occupé.

5-2 Les "features"

En rajoutant des features à un objet (par exemple un personnage) on y rajoute des attributs et des capacités. Souvent, pour pouvoir utiliser certaines commandes comme "@join", think, etc. il faut rajouter une "feature". Dans chaque MOO le numéros des objets changent. Il faut donc demander aux autres le numéro des "features" qui vous intéressent.

La commande help #<no> indique (si disponible) comment l'utiliser.

Exemples sur TECFAMOO:

(1) @addfeature #122

Rajoute une capacité "CB", faire help #122 pour plus d'explications

Liste des features sur TecfaMOO (du 22/12/94)

- #122 xymox's Channel system
- #225 xy's improved social FO
- #227 Stage Talk FO
- #228 Programmer's HELP

- #229 Just for FUN FO
- #248 Not a Silly Social FO
- #310 InterMOO FO (rpage, rwho, etc.)

6. La construction (Building)

6-1 Introduction

Sur la plupart des MOO, un “player” (c’est à dire l’utilisateur “standard”) possède le droit de construire un nombre limité d’objets (souvent il s’agit de 7). En règle générale, on commence par construire une “salle” (room) (un bureau, une maison ou qc de plus fantaisiste), c’est-à-dire un lieu qui sert de maison. Ce room est construit normalement dans le “limbo”, il est déconnecté du reste et doit être connecté par le propriétaire d’un room faisant partie du “réseau MOO”.

Les utilisateurs possédant un “programmer bit” (à demander à un wizard) peuvent créer des objets plus sophistiqués.

Notez que vous pouvez décrire et renommer chaque objet que vous avez crée! Décrire un “room” par exemple est essentiel.

6-2 Commandes de base pour construire

@dig **construction d’un “room”**

Exemples:

(1) @dig palace

“Creuse” un “room” nommé “palace

(2) @dig n,north|s,south to “maison à côté”

Crée d’abord un nouveau room “maison à côté”, puis rajoute une sortie (“exit”) nommé “n” (avec l’alias “north”) depuis le room courant vers le room nouveau, et une sortie “s” depuis le nouveau room vers le room depuis lequel vous “creusez”. Cette commande est pratique pour construire rapidement une zone.

(3) @dig out to #312

Crée une sortie vers le room possédant le No. #312. Notez que vous devez posséder l’objet #312. Sinon il faut utiliser les commandes @add-exit et @add-entrance.

@add-exit

Exemple:

@*add-exit #315*

Rajoute l’ “exit” #315 comme “sortie” du “room” courant. Utilisé pour permettre à qq de lier votre “room” vers le sien en passant par un “exit” que l’autre personne à crée.

@add-entrance

Exemple:

@*add-entrance #316*

Rajoute l’ “exit” #316 comme “entrée du “room courant. Utilisé pour permettre à qq de lier son “room” vers le votre en passant par un exit que l’autre personne à crée.

@remove-exit

Exemple:

@*remove-exit #316*

Enlève la sortie #316 de la salle courante.

@remove-entrance

Exemple:

@*remove-entrance #316*

Enlève l’entrée #316 de la salle courante.

@sethomeDéfinit votre “home”

Exemple:

@*sethome*

Définit le home dans le “room” courant. Notez que vous devez avoir l’autorisation de faire un home ici par le propriétaire du “room”. Il peut le faire avec la commande “@resident <personne>”. On vous voit dans votre home (asleep) lorsque vous êtes déconnectés. La commande “home” vous téléporte chez “vous” en cas de nécessité.

@def **Accès au dictionnaire (US Anglais) pour vérifier un mot, cf “help me:@def”**

Exemple:

(1) @def Rumpelstielzkin

Dans TecfaMOO, les sorties d’une salle sont automatiquement affichées. Pour empêcher ce mécanisme il faut utiliser la techniques suivante: “@set <exit>.obvious is 0”. Obtenez d’abord le No d’objet de la sortie en question (par exemple avec la commande “@examine <exit>, ensuite, par exemple @set #1001.obvious=0.

@describe Décrire un objet

Exemple:

(1) @describe here as “A lovely park with thousands of pink flowers”

Décrit le “room” courant (à condition qu’il vous appartient), Avec la commande ‘@notedit here’ vous pouvez faire la même chose avec l’éditeur de texte interne au MOO.

@rename Donner un autre nom à un objet

Exemple:

(1) @rename me to “débutant”

Vous transforme en débutant ...

(2) @rename #500 to “Porte rouge”, sortie, exit

Renomme la sortie #500 (si elle existe) en “Porte rouge” avec 2 alias.

6-3 L’édition de texte

L’édition de textes associés aux objets est plus ou moins simple suivant le client MOO que vous utilisez. La commande ‘@notedit <quelque chose>’ vous lance (par default) dans l’éditeur interne du MOO. Faites ‘look’ une fois que vous êtes dedans. Il s’agit d’un simple éditeur qui vous permet d’éditer ligne par ligne.

Quelques commandes de base:

list **Afficher les lignes d’un text**

save **Sauvegarder les modifications**

quit **Quitter l’éditeur**

“ **Inséerer une ligne de texte à l’endroit où se trouve le pointeur**

(1) “this is the first line

Insère une ligne “this is the first line” (à l’endroit où se trouve le pointeur)

ins **Insérer une nouvelle ligne**

(1) ins 3 “this text is nice

Insère une nouvelle ligne 3 avec le contenu “this text is nice”.

del **Détruire une ligne**

(1) del 3

détruit la ligne No. 3

(2) del 4-10

détruit les lignes 4-10.

Pour les gens qui utilisent un client emacs, il est conseillé de taper ‘@edito +l’ (une seule fois), dans la suite toute édition (mail, textes, etc.) vous ouvre un “buffer”. Ne détruisez pas la première ligne, ni le “.” à la fin et insérez votre texte entre ces lignes. Pour sauvegarder votre édition, tapez “Ctrl-C Ctrl-S”).

6-4 Gestions de ses objets

@audit **Lister tous les objets en possession**

@quota Afficher les quota disponible

@count Afficher le nombre d’objets en possession

@recycle Détruire un objet (attention!)

Exemple:

(1) @recycle #301

Détruit l’objet No.1

@move Bouger un objet

(Utile par exemple pour récupérer un objet perdu ou volé)
Exemple:

(1) move #301 to here

Amène l’objet #301 dans la salle courante.

6-5 Description plus avancée de “rooms” et sorties

En règle générale, on attache un certain nombre de messages aux entrées et sorties pour rendre le passage plus intéressant.

@messages imprime les messages attachables à un objet

Exemple:

(1) @messages me

Indique la liste des messages attachés à votre personnage

(2) @messages <exit>

Indique la liste des messages attachés à un “exit”

<@message-name> <object> is <“message ....”>

Exemple:

(1) @leave n is “Vous vous promenez vers le nord ..... “

Ce message est vu par la personne qui emprunte une sortie. Utilisé à divers raisons, par exemple pour décrire une marche plus longue (par ex. “vous prenez le bus et .....”) ou encore pour rendre la sortie intéressante (par ex “vous évitez de justesse de vous cogner en sortant...”).

(2) @oleave n is “struggles towards the north”

Message destiné aux gens qui restent dans la même salle.

Mise en page de messages

1. Tous les noms des messages commençant par un “o” produisent un message qui met le nom de l’utilisateur au début du message
2. Utilisation de variables: Dans un string vous mettre des variables qui seront substitués par des noms d’objets ou des promons (en Anglais). Faites un “help messages” pour des détails, ici juste le résumé en Anglais:

```
    %% => `%'  (just in case you actually want to talk about percentages).
Names:
    %n => the player
    %t => this object (i.e., the object issuing the message,... usually)
    %d => the direct object from the command line
    %i => the indirect object from the command line
    %l => the location of the player
Pronouns:
    %s => subject pronoun:          either `he',  `she', or `it'
    %o => object pronoun:           either `him', `her', or `it'
    %p => posessive pronoun (adj):  either `his', `her', or `its'
    %q => posessive pronoun (noun): either `his', `hers', or `its'
    %r => reflexive pronoun:  either `himself', `herself', or `itself'
General properties:
    %(foo) => player.foo
    %[tfoo], %[dfoo], %[ifoo], %[lfoo]
           => this.foo, dobj.foo, iobj.foo, and player.location.foo
Object numbers:
    %#  => player's object number
    %[#t], %[#d], %[#i], %[#l]
       => object numbers for this, direct obj, indirect obj, and location.
```

```
In addition there is a set of capitalized substitutions for use at the
beginning of sentences.  These are, respectively,
```

```
%N, %T, %D, %I, %L for object names,
%S, %O, %P, %Q, %R for pronouns, and
%(Foo), %[dFoo] (== %[Dfoo] == %[DFoo]),... for general properties
```

Exemples:

(1)

6-6 Création d’objets à partir d’objets génériques

‘help building’ vous lance dans le help interne du MOO (valable dans tous les MOO).

@create Création d’objets (autres que “rooms” et “exits”)

@create <objet-parent> named “<noms>

Exemples:

(1) @create <#xxx> named “au hasard”

Cré un “enfant” de l’objet #xxx

@create <nom de classe> named “<noms>”

Exemples:

(1) @create $note named “affiche”

Crée un objet sur lequel on peut écrire (avec nom = “affiche”)

@lock Vérouiller un objet

Exemple:

(1) @lock “Grande affiche” with me

Interdit aux autres de prendre votre affiche.

6-7 Les objets génériques du TecfaMOO

Sur chaque MOO il existe un certain nombre d’objets génériques que vous pouvez utiliser pour construire quelque chose. Cf. par exemple la liste des objets standards avec ‘@classes generics’.

On peut retrouver les autres objets disponibles en fouillant la liste des objets, toutefois il est plutôt conseillé de lire les “news” interne du MOO, par exemple faites ‘@mail on *new*’ (cf . la section 2-4 “Communication asynchrone/Systèmes d’information” [p. 5]. Sinon consultez le MOOseum où on met les objets les plus intéressants en display.

Le MOOseum

Le MOOseum (#258) est un endroit où on met en display les objets génériques les plus intéressants.

Quelques objets intéressants du TecfaMOO (à faire, cf. le MOOseum pour le moment)

1. Les objets à usage pédagogiques
2. Les “poupées”
3. le “noisy room”
4. le “portable room”
5. le “generic notebook”

7. Extensions WWW<->MOO

Le client WWW interne

Il faut d'abord se construire un client à soi que l'on porte avec soi avec la commande:

@create $www_slated named <"nom que vous lui donnez">[<,"alias"\

Ensuite: help <"nom de votre client">

Quelques commandes (on assume que vous avez appelé votre slate "monwww":

(1) goto http://tecfa.unige.ch/ 80 on monwww
Vous connecte sur le serveur www du tecfa
(2) pop monwww
retourne vers le "haut" (un niveau, pour plus de niveaux utiliser: pop monwww for x
(3) pick 2 on monwww
affiche la page no2 sur monwww
(4) remember 3 on monwww
se souvent de ce noeud
(5) show monwww to julie
montrer le contenu du slate a julie

Un client WWW->MOO

Pas disponibles (pendant un moment encore) sur TecfaMOO. Il existe plusieurs prototypes, dont l'interface à BioMOO:

1. Se connecter comme guest
2. @password
3. Lancer NCSA Mosaic et ouvrir http://bioinfo.weizmann.ac.il:8000/ et rentrer le mot de passe du MOO
4. Ensuite il faut naviger dans le Web (juste discuter dans le MOO)