

Automatisation de tests HandOver



Table des matières

.....	1
I. Avant-Propos.....	1
Introduction.....	2
II. Quelques Etapes préliminaires.....	3
III. Installation de Python2.7	6
IV. L'exécution du programme	7
A. Les différents types D'appels	8
1. CALL PTP	10
2. CALL DATA	10
3. GROUP CALL (VGCS, VBS, REC)	10
B. Le mode Simple	11
C. Le mode Reset-TMU	12

I. Avant-Propos

- Ce document a pour objectif de vous guider dans l'utilisation de l'application et de répondre aux éventuelles questions que vous pourriez vous poser au cours de son usage
- Les détails, les données, les résultats etc.... inclus dans ce document ont été écrits et vérifiés au mieux des connaissances de l'auteur. Néanmoins, des erreurs dans le contenu sont possibles.
- L'application automatisée de test a été réalisée pour répondre à certains besoins spécifiques de Kapsch Carrier Com France. Ceci signifie qu'il est important de lire le contenu de ce manuel d'utilisateur avant de commencer à l'utiliser.

Introduction

La mise en place cet outil a pour but d'améliorer les méthodes de test en remplaçant les tests manuels par des tests automatisés. Cet outil permet dans les différentes campagnes de tests de vérifier automatiquement l'intégrité des fonctionnalités du système et de détecter les régressions éventuelles. En générant automatiquement un fichier de rapport beaucoup plus facile à comprendre, et qui contient la quasi-totalité des informations reçues pendant le déroulement d'un test.

Le programme est aussi capable de gérer certaines erreurs courant, qui sont des erreurs intentionnelles ou non intentionnelles d'un utilisateur, ou d'autres types d'erreurs fréquentes telles que : les problèmes avec les mobiles, les erreurs sur les réseaux (erreurs de bourrages, de pertes de données...). De plus, il possède une interface homme-machine (IHM) pour avoir une interaction simple avec l'utilisateur, ce qui lui rend donc maniable. Son interface graphique est uniforme pour plus de souplesse.

L'outil possède 5 familles de tests qui sont les HandOver en mode PTP, DATA, VGCS, REC, VBS. Chaque scénarios doit se faire avec en précisant le type d'option en Handover c'est-à-dire le mode simple ou un mode reset TMU.

Le Point fort de ce outil c'est qu'il est d'automatiser des HandOver Inter BSC ou interMSC ce qui est beaucoup plus intéressant.

II. Quelques Etapes préliminaires

Cette section présente certains éléments à prendre en considération avant de dérouler un call. Ces actions sont des contraintes indispensable pour le bon fonctionnement des calls CSD.

- Le Redémarrage du mobile : Dans la plus part des cas les mobiles sont à la base du dysfonctionnement de certains tests. Alors il est préférable de redémarrer les deux mobiles avant de faire toute chose.
- Pour les appels groupe comme VGCS, VBS, REC, j'ai pas trouver une command AT compatible avec les mobiles Tijrs 555r pour accepter ou raccrocher un group call. Alors j'ai envisagé de choisir un dispatcher avec la place de l'un de mobile car les dispatcher sont capable d'accepter de façon automatique les groupe calls, et ils ont la possibilité de kill le groupe call envoyé trois Etoiles *** en mode DTMF. Cependant, il est donc conseillé d'avoir à sa possession un dispatcher, ou un mobile configuré avec comme dispatcher possédant une carte SIM bien reconnue comme dispatcher. Car cela permet au mobile de pouvoir accepter un groupe call de façon automatique ou la capacité de raccrocher un groupe call sans aucune innervation humaine. Par contre pour les call PTP ou DATA, le programme n'a pas forcément besoin d'un dispatcher, deux mobiles bien faire l'affaire.
- Identification des Mobiles : L'identification des mobiles est l'étape la plus importante, il faudra être sûr que le mobile est bien reconnue par la machine, identifié la ou les COM qui correspondent aux éventuels mobiles et connaitre quels mobiles devraient correspondre au mobile émetteur et récepteur et aussi le mobile de transfert d'appel dans d'autre cas.

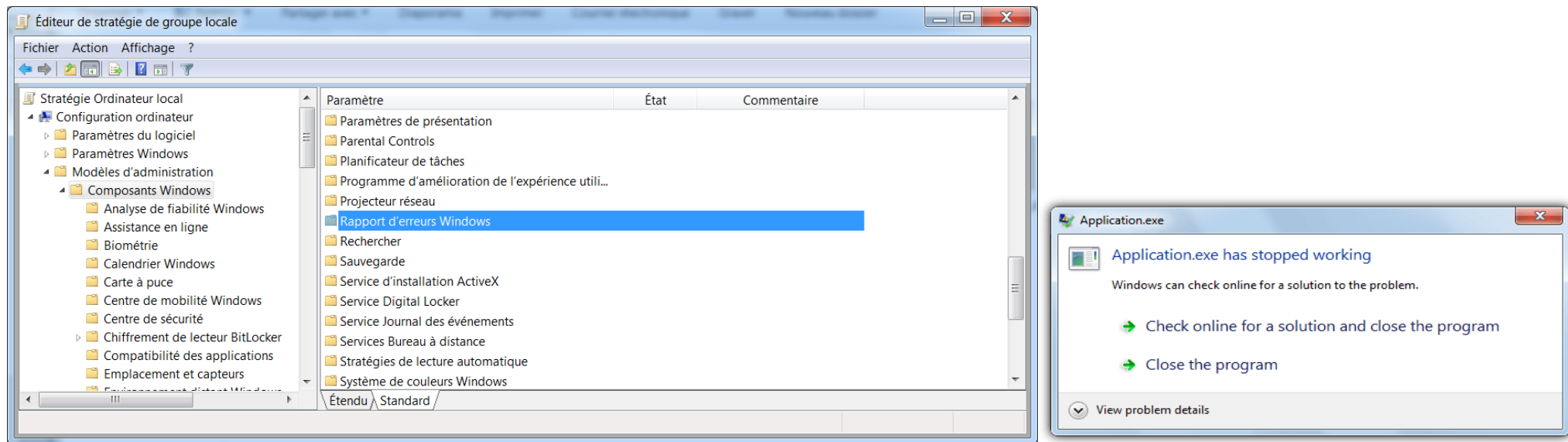
Outil de tests automatiques

■ Configuration du système d'exploitation de windows 7 :

Ce n'est pas idéal lorsqu'on utilise des scripts d'automatisation de longue durée sans intervention de l'utilisateur. Le problème est que Windows stop l'exécution du script par l'apparition d'une fenêtre. Cette fenêtre de vérification fait partie du rapport d'erreur de Windows. Sa fonction est de recueillir et d'envoyer des données à Microsoft et si elle a une solution possible de l'incident alors, il est envoyé à l'utilisateur.

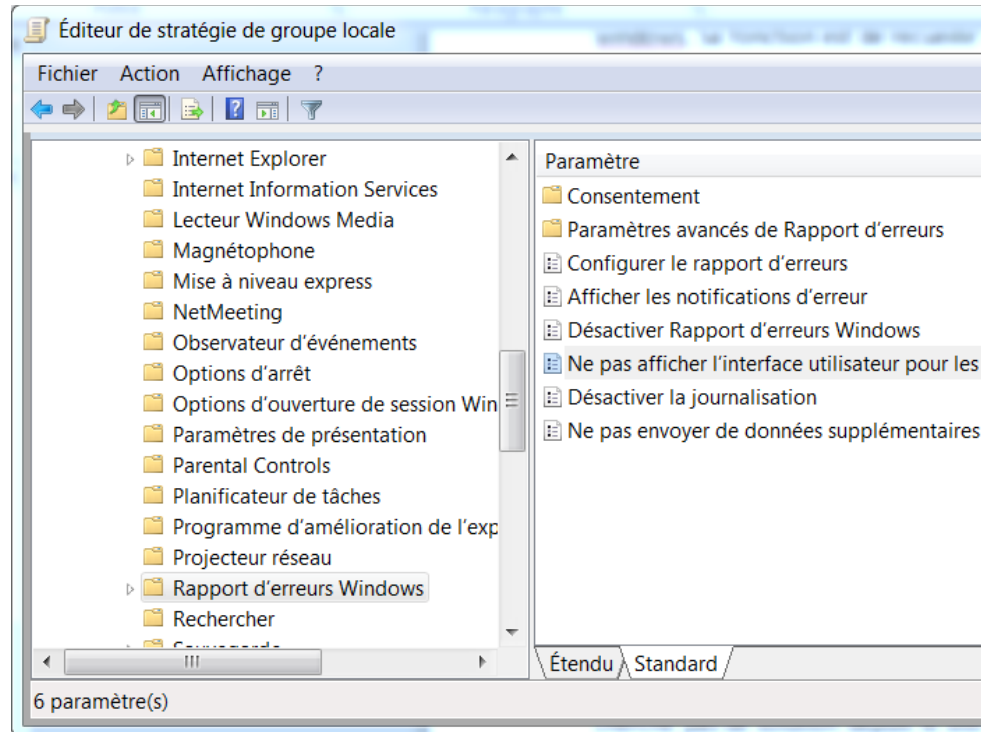
La solution est de désactiver complètement la fenêtre par une modification manuelle du registre GPEdit. Cette modification est très utile pour le long CSD.

- Ouvrez le registre dans le menu Démarrer en tapant « gpedit.msc » dans la zone de recherche.
- Accédez à la configuration ordinateur > Modèles d'administration> Composants Windows > Rapport d'erreurs Windows

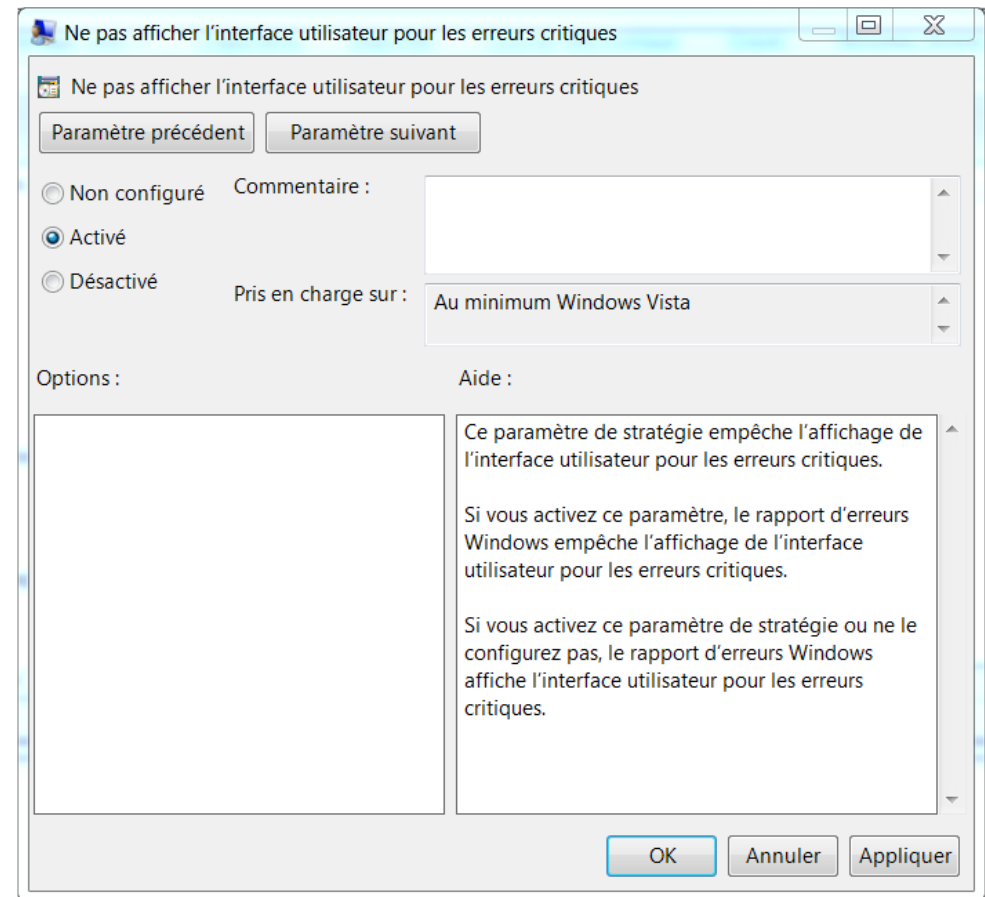


Outil de tests automatiques

- Double-cliquez sur « Ne pas afficher l'interface utilisateur pour les erreurs critiques » dans le volet de droite



- Cliquez sur le radio bouton « Activé » puis appuyez sur « Ok »

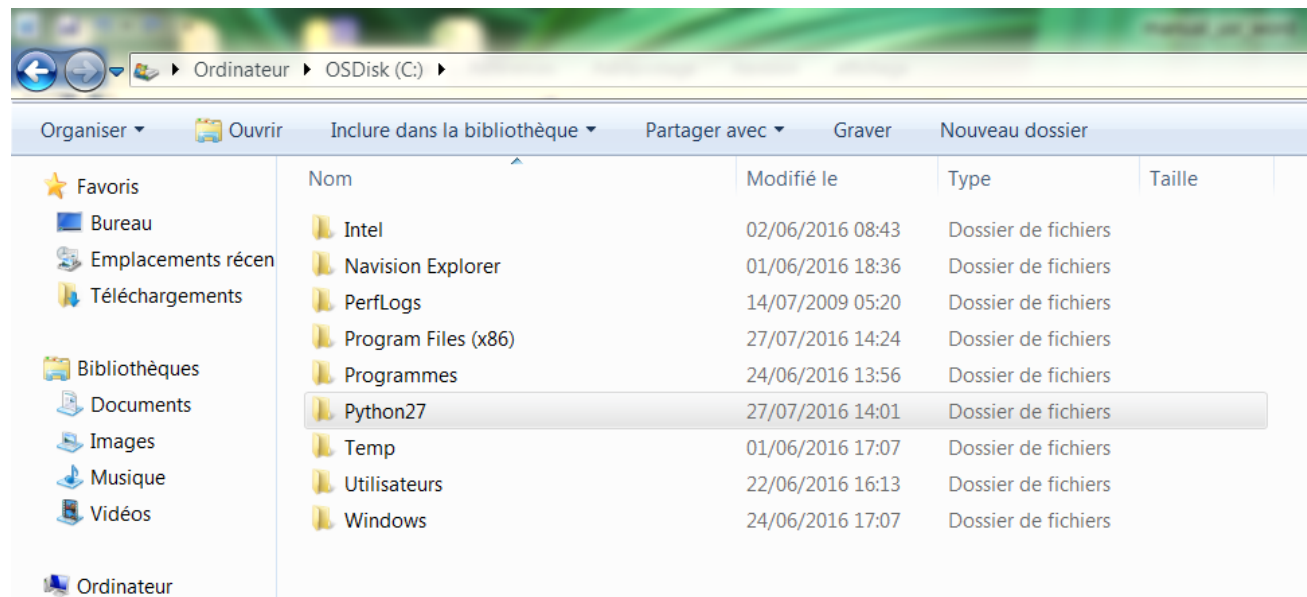


III. Installation de Python2.7

Vous trouverez un fichier **Python2.7.zip** dans le répertoire du programme. Ce qui vous évitera de faire des recherches pour l'installation les différents packages.

L'installation de python2.7 se fait en plusieurs étapes :

- La première étape est de dézipper le fichier python2.7.zip qui se trouve dans le dossier du programme de test.
- Faire une copie du dossier python2.7 et ensuite place le dans le répertoire C:\ (voir ci-dessous)



- Le fait de placer le dossier python 2.7 dans le répertoire racine ne veut pas dire forcément que python est installé sur la machine. Donc pour finaliser l'installation vous devez télécharger l'exécutable de python2.7 depuis le site officiel et l'installer sur votre machine : <https://www.python.org/downloads/release/python-2711/>.

IV. L'exécution du programme

Pour lancer le programme vous avez deux choix possibles :

- La premier est de faire un clic droit sur le fichier setup.py et choisir d'ouvrir avec python.
- Pour la deuxième option, allez dans le menu Démarrer et saisir cmd dans la zone de rechercher.
 - Placez-vous dans le répertoire qui contient le programme à l'aide de la commande cd
 - Ensuite lancer le programme avec en précision le chemin du répertoire python.

Par exemple pour mon cas le répertoire de python est : c:\Python27\python.exe donc pour lancer il vous suffit de faire :

```
C:\Users\IT MTY\Desktop\handover\hand_prog>  
C:\Users\IT MTY\Desktop\handover\hand_prog>  
C:\Users\IT MTY\Desktop\handover\hand_prog>C:\Python27\python.exe Setup.py
```

Si tout se passe bien l'apparition d'une fenêtre doit vous permet de savoir que le script est lance

A. Les différents types D'appels

Outil de tests automatiques

CONFIG HandOver

HandOver Type

- ☒ SIMPLE
- ☐ RESET-TMU
- ☐ PTP-PCM

Type of call

- ☐ PTP
- ☐ DATA
- ☒ VGCS
- ☒ REC
- ☒ VBS

Choisir un port COM

COM mobile phone: COM4

COM Dispatcher: COM5

Num Dispatcher: 555558000265

Cmd AT+ CSBT: 70,0,0

Groupe ID: 200

NB Reset TMU: 2

Config OMC

Hostname	Username	Password
172.21.155.71	omc	nortel

Config BSC1

Hostname	Username	Password
47.164.27.128	omu	omu

Config BSC2

Hostname	Username	Password
47.164.27.144	omu	omu

Exécute

CONFIG HandOver : Cette partie permet de choisir un HandOver avec Reset-TMU ou un HandOver SIMPLE. Pour le RESET-TMU elle permet de se connecter à l'OMU reseté l'ensemble des TMUs. Par contre avec le mode SIMPLE vérifié que la conf est capable de faire un HandOver. Et elle se limite donc à un seul HandOver.

Type of Call : c'est le type d'appel qu'il faudra choisir pour faire un HandOver

COM Mobile phone : c'est le com du mobile émetteur.

COM Dispatcher : c'est le com qui correspond au mobile dispatcher. Pour les calls du types VGCS, VBS, REC, il est conseillé d'utiliser un dispatcher pour que le programme est la possibilité de killer ou d'accepter un groupe call. Par contre les calls PTP ou Data on ne pas mettre un dispatcher, un mobile normale fait l'affaire.

Num Dispatcher : c'est le numéro du dispatcher ou le mobile qui reçoit l'appel. Attention ce numéro doit être différent de celui mobile qui émet l'appel.

Cmd AT+ CSBT : c'est la configuration qui doit être utilisée quand le programme utilise un HandOver de type DATA. Ce champ permet au mobile d'avoir la même vitesse.

GroupeID : c'est l'identifiant du groupe call (VGCS, VBS), pas besoin de saisir pour le REC car il est préalablement configuré dans le programme.

NB Reset TMU : le nombre de fois où chaque TMU du BSC doit être reseté : par exemple si un BSC on 5 TMUs, et si le NB Reset TMU est égale à 2 alors chaque TMU du BSC sera reseté deux fois de façon aléatoire.

Config OMC : contient l'adresse IP, Username et password de l'OMC

Config BSCx : C'est les identifiants des différents BSC. Pas d'ordre spécifique.

1. CALL PTP

Il s'agit ici d'un mode d'appel très utilisé dans la téléphonie. Une conversation avec une personne qui est loin, plus exactement, trop loin pour une conversation de vive voix. L'appel est bidirectionnel : les deux personnes en peuvent se parler et s'entendre en même temps, comme dans une conversation face à face. Ainsi, le programme doit pouvoir simuler cette possibilité.

2. CALL DATA

L'établissement d'un mobile à commutation de circuit de données se fait avec l'option DATA (call CSD). Cet appel fonctionne en full-duplex, et la liaison RF est maintenue et se termine lorsqu'on est sûr que les deux mobiles ont fait le HandOver, ou bien s'il y a une exception qui se produit. Cependant il y a deux façons d'utiliser le HandOver avec le call. Pour cela, il faut choisir soit le mode SIMPLE ou le mode avec Reset-TMU.

3. GROUP CALL (VGCS, VBS, REC)

Le groupe Call est une communication de type d'émetteur-récepteur de radiotéléphonique. Une liaison half-duplex, c'est-à-dire que la communication se passe dans les deux sens, mais pas simultanément. Chaque interlocuteur parle à tour de rôle en appuyant ou en maintenant un bouton poussoir, et en le relâchant dès qu'il a fini de parler. L'autre peut alors répondre en effectuant la même action.

Pour exécuter un test du groupe call, vous devez choisir le port COM correspondant du mobile qui doit établir un groupe Call, et de saisir dans un champ texte **le groupe ID**.

B. Le mode Simple

Pour les tous appels DATA, PTP, VGCS, VBS et REC le principe d'exécution reste toujours le même. Le HandOver avec call en mode SIMPLE consiste à établir un appel et de vérifier qu'il y'a un HandOver.

Pour cela, il fait call CSD et attend de voir qu'il y'a un connect. Il se connecte ensuite sur L'OMC pour lancer une commande qui retourne la liste des call processus busy sur chaque BSC. Alors le programme compte le nombre de busy de chaque côté.

Ensuite il exécute le fichier de command qui se trouve dans le répertoire **/OMC/data/cmdFile/root/users/fhissirou/Force_HO_BSC17** ou **/OMC/data/cmdFile/root/users/fhissirou/Force_HO_BSC140** selon le type de BSC. Les commandes contenues dans ce fichier permet de faire un Force HandOver. Alors, tous les mobiles présent dans ce cellule se retrouve dans un autres cellule. Une fois que cette tâche effectué. Le programme vérifie à nouveau le nombre de call Bussy. Si le nombre de Busy est différent de celui du nombre premier alors il a un Hanover.

Donc il y'a au moins un mobiles qui est parti sur une autres cellule. Dans le cas contraire il en déduit qu'il n'Ya pas un errer et qu'il n'y a pas de Hanover.

C. Le mode Reset-TMU

Dans le cas d'un call avec Reset-TMU, le principe est similaire à celui de de monde. La seule différence c'est que les mobiles reset les TMU.

En mode reset TMU une fois qu'un call est en cours, le programme se connecte sur l'OMU pour récupérer la liste des TMU disponible. Il est important de souligné que lorsqu'un TMU est en panne il donc pas répertorier dans la liste.

Il choisit donc aléatoirement TMU pour reset dans chacune des BSC. Le reset d'un TMU peut prendre plusieurs minutes environ 120s et cela prend un peu de temps. Le programme est donc autorisé à faire un tempo de 160s. Ensuite il vérifie l'existence d'un HandOver comme celui de l'étape précédant avec le mode SIMPLE. Il choisit un autre TMU aléatoirement pour reset. Il bon de savoir que chacune TMU est reseter au moins une fois. En fonction du nombre de reset max mal choisi par l'utilisateur.

D. Le mode Reset-TMU

Au côté OMC, tous les fichiers doivent être sauvegardé dans le répertoire :

/OMC/data/cmdFile/root/users/fhissirou/

Dans ce répertoire, vous devez avoir 4 fichiers différents avec comme : **Force_HO_BSC17, Force_HO_BSC140, Display_CHANNEL_BSC17, Display_CHANNEL_BSC140.**

Attention vous ne devez surtout pas vous trompez des leurs noms.

1. Contenu du fichier Force_HO_BSC17

```
1  CMDFILE_OMC_VERSION:
2  OWNER:oam
3  CREATIONDATE:07/27/17 17:18:30
4  MODIFICATIONDATE:07/27/17 17:18:30
5  AUTHORIZEDUSER:
6  SUMMARY:
7  END_OF_CMDFILE_HEADER
8  Set-handOverControl: object = (bsc = 17 & btsSiteManager = 0 & bts = 1 & handOverControl = 0); lRxLevDLH = "less than -110";
9  Force HandOver-bts: object = (bsc = 17 & btsSiteManager = 0 & bts = 1);
```

2. Contenu du Fichier Force_HO_BSC140

```
1  CMDFILE_OMC_VERSION:
2  OWNER:oam
3  CREATIONDATE:07/27/17 17:18:30
4  MODIFICATIONDATE:07/27/17 17:18:30
5  AUTHORIZEDUSER:
6  SUMMARY:
7  END_OF_CMDFILE_HEADER
8  Set-handOverControl: object = (bsc = 140 & btsSiteManager = 0 & bts = 0 & handOverControl = 0); lRxLevDLH = "less than -110";
9  Force HandOver-bts: object = (bsc = 140 & btsSiteManager = 0 & bts = 0);
```

3. Contenu du fichier Display_CHANNEL_BSC17

```
1  CMDFILE_OMC_VERSION:
2  OWNER:oam
3  CREATIONDATE:07/27/17 17:18:30
4  MODIFICATIONDATE:07/27/17 17:18:30
5  AUTHORIZEDUSER:
6  SUMMARY:
7  END_OF_CMDFILE_HEADER
8  Display channel state-transceiver: object = (bsc = 17 & btsSiteManager = 0 & bts = 1 & transceiver = 0);
9  Display channel state-transceiver: object = (bsc = 17 & btsSiteManager = 0 & bts = 1 & transceiver = 1);
10 Display channel state-transceiver: object = (bsc = 17 & btsSiteManager = 0 & bts = 1 & transceiver = 2);
```

4. Contenu du fichier Display_CHANNEL_BSC140

```
1  CMDFILE_OMC_VERSION:
2  OWNER:oam
3  CREATIONDATE:07/27/17 17:18:30
4  MODIFICATIONDATE:07/27/17 17:18:30
5  AUTHORIZEDUSER:
6  SUMMARY:
7  END_OF_CMDFILE_HEADER
8  Display channel state-transceiver: object = (bsc = 140 & btsSiteManager = 0 & bts = 0 & transceiver = 0);
9  |
```