

Documentation technique MedPlaus 6

(S. Freudiger, S. Hasler)

18 janvier 2023

Freudiger EDV-Beratung
Zeughausgasse 16
3011 Berne
Suisse

www.freudiger.com

Table des matières

1	Introduction	3
1.1	Versions	3
1.2	Téléchargement	3
2	MedPlaus stand alone	4
3	MedPlaus Shell	5
4	MedPlaus DLL	6
4.1	Installation	6
4.1.1	Compatibilité avec MedPlaus 5	6
4.1.2	Sous-répertoires	6
4.1.3	Mises à jour	7
4.2	Contenu de la livraison	7
4.3	Types de messages	8
4.4	Tests	9
4.4.1	Nouvelle hiérarchie des tests/numéros des tests	9
4.4.2	Concept de plausibilisation	9
4.5	Description des fonctions	10
4.5.1	Encodage des jeux de caractères	10
4.5.2	Description des types	10
4.5.3	function MP_Create()	10
4.5.4	function MP_setFormatString()	11
4.5.5	function MP_TestRec()	12
4.5.6	function MP_GetMessageList()	12
4.5.7	function MP_Destroy()	13
4.5.8	function MP_VersionAppName()	13
4.5.9	function MP_VersionNumber()	13
4.5.10	function MP_VersionDate()	13
4.6	MP6 DLL-DynTest	14
4.6.1	Préparations du test	15
4.6.2	Conduite du test	16
4.6.3	Informations supplémentaires	18
4.7	Recherche d'erreurs	20
5	Divers	21
5.1	Support par le fabricant	21
5.2	Historique du document	21
5.3	Version	21

1 Introduction

MedPlaus 6 est le logiciel officiel pour la plausibilisation des données de la statistique médicale des hôpitaux. Il sert à tester l'exactitude des enregistrements et exportations OFS.

MedPlaus est développé et mis à disposition sur mandat de l'Office fédéral de la statistique (OFS). Le logiciel fonctionne avec toutes les versions Windows actuelles.

L'application contient des centaines de règles, lesquelles permettent de vérifier les informations de l'enregistrement minimal, y compris les codes des diagnostics (CIM) et les codes des opérations (CHOP). Les enregistrements supplémentaires consacrés aux nouveau-nés et à la psychiatrie sont également testés, de même que l'enregistrement MD.

MedPlaus 6 est optimisé quant à sa vitesse. Même de très volumineux fichiers sont rapidement contrôlés.

1.1 Versions

Il existe trois versions de MedPlaus :




Version	MedPlaus stand alone	MedPlaus Shell	MedPlaus DLL
			
Pour	utilisateurs finaux	poweruser	fabricants de logiciels
Description	Elle contient un programme d'installation et une interface graphique pour les utilisateurs. Grâce à cet outil, les collaborateurs dans les hôpitaux et les administrations peuvent contrôler la qualité des exportations trimestrielles et annuelles.	Grâce à son exceptionnelle rapidité, la version ligne de commande de MedPlaus constitue un outil idéal pour l'évaluation et la plausibilisation de vastes stocks de données.	Elle permet l'intégration de la fonctionnalité de MedPlaus dans des outils produits par des tiers. Avec la DLL, des enregistrements séparés peuvent être contrôlés.
Livraison	fichier d'installation (mp600ch.exe)	fait partie de la distribution MedPlaus DLL (fichier zip)	distribution MedPlaus DLL (fichier zip)

FIGURE 1.1 – Versions de MedPlaus

1.2 Téléchargement

Les distributions peuvent être téléchargées à partir de la page MedPlaus de la Freudiger EDV-Beratung. Vous la trouverez sous :

- <http://www.freudiger.com/f/medplaus/> (Page d'accueil en langue française)
- <http://www.freudiger.com/medplaus/> (Page d'accueil en langue allemande)

2 MedPlaus stand alone

La version stand alone n'est pas décrite dans ce document. Elle contient une aide en ligne très détaillée. Cette dernière peut être lancée à partir du programme, une fois l'installation réussie.

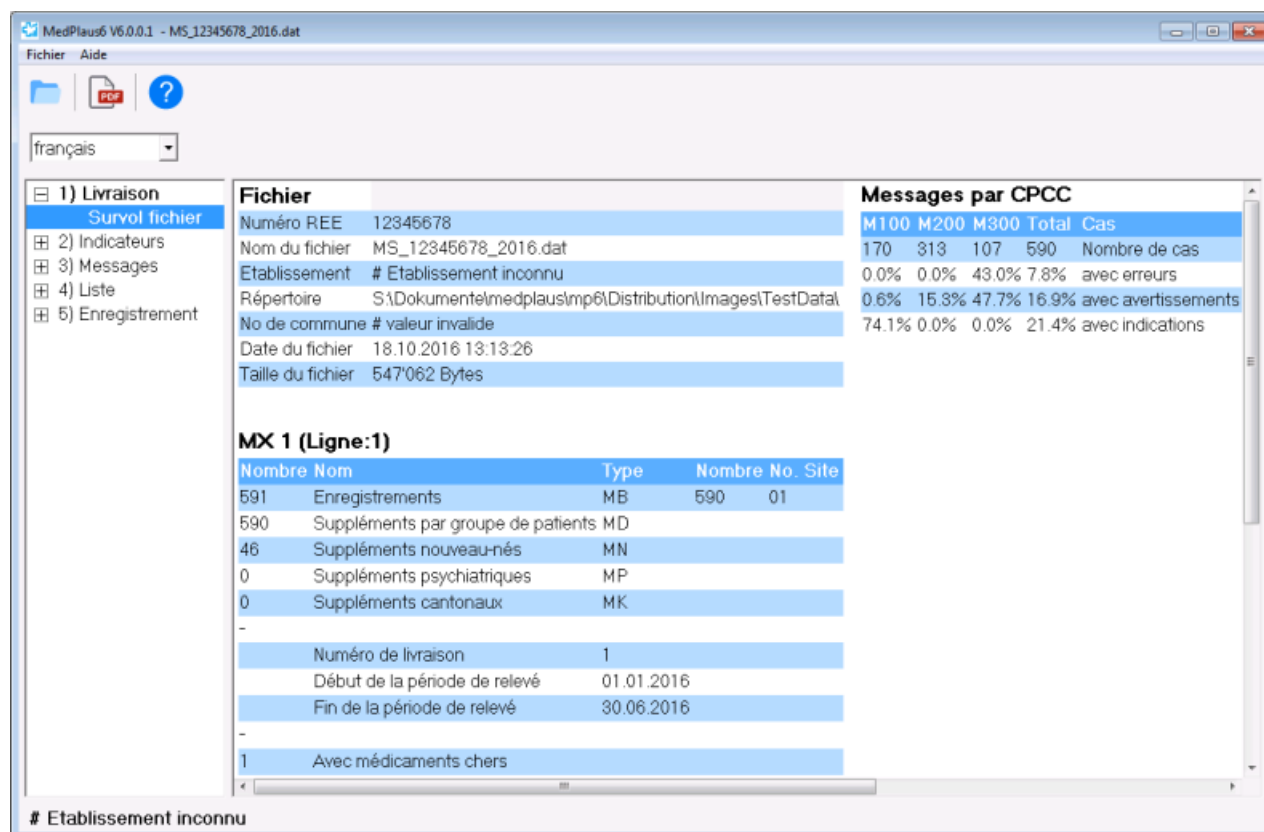


FIGURE 2.1 – MedPlaus 6 stand alone

3 MedPlaus Shell

MedPlaus peut être démarré, avec de nombreux paramètres, par ligne de commande. Par exemple en tant que batch.

L'appel est structuré de la manière suivante :

— `medplaus6.exe [Options] [DataFilename]`

L'appel seul de `medplaus6.exe` liste tous les paramètres possibles.

Il existe les [Options] suivantes :

Paramètre	Signification
-de	Crée les textes allemands pour les messages log (par défaut)
-fr	Crée les textes français pour les messages log
-log=FILENAME	Ecrit des messages dans le fichier FILENAME
-log	Ecrit des messages dans le fichier DataFilename.log
-report=FILENAME	Ecrit un rapport MX dans le fichier FILENAME
-report	Ecrit un rapport MX dans le fichier DataFilename.rep
-stat=FILENAME	Ecrit des valeurs statistiques dans le fichier FILENAME
-stat	Ecrit des valeurs statistiques dans le fichier DataFilename.stat
-f [formatString]	Utilise un format output personnalisé (voir <code>MP_setFormatString()</code>)
-mppath=PATH	Définit le chemin des fichiers index (*.ix5)
-version	Affiche la version
-MKGR	Intégration de l'enregistrement supplémentaire cantonal des Grisons
-MKLU	Intégration de l'enregistrement supplémentaire cantonal Suisse centrale (LUSTAT)

Les règles et conventions suivantes sont en vigueur :

- L'ordre de préséance des paramètres ne joue aucun rôle
- Les majuscules et les minuscules doivent être prises en considération
- Un chemin devrait être indiqué pour tous les noms de fichier. Sans quoi, le répertoire en vigueur est celui duquel provient l'appel

Exemple : Démarrez le programme avec un fichier, création d'un fichier log en langue française comprise

- Cliquez sur le bouton *Démarrer*
- Sélectionnez *Exécuter*
- Entrez le texte suivant (adaptez, selon vos besoins, les noms des lecteur, répertoire et fichier) :
`c:\programme\mp6070\medplaus6.exe -fr -log=c:\logfiles\2023\test.log test.dat`

4 MedPlaus DLL

La DLL de MedPlaus 6.0.7.0 est, en ce qui concerne les appels des fonctions et à l'exception de la fonction élargie MP_TestRec, de la nouvelle et optionnelle fonction MP_setFormatString() et du changement de noms de quelques fichiers DLL, compatible avec la version précédente MedPlaus 5. Elle nécessite un minimum de 10 Mbyte de mémoire vive, puisqu'elle y charge toutes les informations nécessaires.

Le fichier MedPlaus6DLL.dll est, c'est nouveau, en soi multilingue. Il introduit un paramètre de langue à la fonction MP_TestRec(). Le changement de langue est instantanément possible.

Une version 64 bit de la DLL existe également, y compris l'interface graphique de test MP6 DLL. Le fichier se nomme medPlaus6DLL64bit.dll.

4.1 Installation

Les fichiers de la version DLL sont contenus dans le fichier medplaus_dll_v6070_20230118.zip.

Extrayez les fichiers dans un répertoire et communiquez avec la medplaus6DLL.dll¹ (resp. medplaus6DLL64bit.dll) depuis votre logiciel.

La DLL fonctionne sans installation de MedPlaus 6 stand alone.

4.1.1 Compatibilité avec MedPlaus 5

Si l'on compare les DLL de MedPlaus 5 et MedPlaus 6, l'on voit que cette dernière présente les différences suivantes :

- Nouveau paramètre cLanguage² dans l'appel de la fonction MP_TestRec() de la DLL standard
- Nouveaux noms de fichiers
 - medplaus6DLL.dll au lieu de medplaus5DLL.dll
 - medplaus6DLL64bit.dll au lieu de medplaus5DLL64bit.dll
- Nouvelle fonction MP_setFormatString()
 - Sans appel de celle-ci, le format par défaut est le format jusqu'à présent compatible avec MP5

4.1.2 Sous-répertoires

La documentation se trouve dans le répertoire \help.

Dans le répertoire \DLL.BigLog se trouvent des version alternatives de la DLL. Elles offrent une fonctionnalité LOG complète qui vous aide dans la recherche des erreurs. Le fichier du protocole est écrit dans le répertoire c:\temp. Celui-ci doit déjà exister.

Ces versions sont nettement plus lentes que les DLL standards.

1. Alternative : Lancez medplaus6.exe en mode batch avec les paramètres nécessaires (voir MedPlaus Shell)
2. Faisant jusqu'ici partie de la medplaus5DLL_ML.dll

4.1.3 Mises à jour

En règle générale, dans une même version de MedPlaus, les fichiers de l'actuelle distribution zip peuvent être copiés dans l'environnement existant. Il suffit donc de remplacer les fichiers déjà présents par les nouveaux.

Dans le doute, veuillez consulter la documentation technique. Si celle-ci ne devait pas suffire, contactez la hotline de la Freudiger EDV-Beratung.

4.2 Contenu de la livraison

Les fichiers les plus importants sont :

Nom du fichier	Commentaire
medplaus6DLL.dll	DLL 32 bit, productive et multilingue, de MedPlaus 6
medplaus6DLL64bit.dll	DLL 64 bit, productive et multilingue, de MedPlaus 6
medplaus6DLL_BigLog.dll	DLL alternative 32 bit de MedPlaus 6, avec fonctionnalité LOG complète
medplaus6DLL64bit_BigLog.dll	DLL alternative 64 bit de MedPlaus 6, avec fonctionnalité LOG complète
bur.ix5	Index des numéros REE des établissements
chop.ix5	Index CHOP (version 2023 et versions précédentes)
icd.ix5	Index CIM-10 GM (version GM 2022 et versions précédentes)
natio.ix5	Index des nationalités
plz.ix5	Index des NPA
regio.ix5	Index des codes des régions suisses
wregio.ix5	Index des régions du monde
mp6show.exe	Fenêtre d'affichage séparée pour un enregistrement et les messages provenant de la mp6DllDynTest.exe et mp6DllDynTest64.exe
mp6DllDynTest.exe	GUI pour tester les commandes de la DLL (32 bit)
mp6DllDynTest64.exe	GUI pour tester les commandes de la DLL (64 bit)
mp60_dll_changes_f.pdf	Documentations des changements de la MedPlaus DLL indiquée (version pdf)
mp6070_dll_techDoc_f.pdf	TechDoc pour la MedPlaus 6-DLL
MS_Schnittstellenkonzept_2020_f.pdf	Conception d'interface 2020 de l'OFS

Dans MedPlaus 6, les fichiers index continuent de porter l'extension ix5 (le nouveau format qui existe depuis MedPlaus 5).

4.3 Types de messages

MedPlaus 6 connaît les messages suivants :

Type de message	Commentaire
Erreur	Les erreurs doivent être corrigées. Taux de tolérance admis par l'OFS : max. 1%.
Avertissement	Devrait être vérifié/corrigé. Taux de tolérance admis par l'OFS : max. 50%.
Indication	Livre des informations suppl. permettant une amélioration de la qualité des données.
Structure	Suggère des erreurs structurelles (par ex. un enregistrement ou un caractère avec format inadéquat). Dans un tel cas, la valeur transmise en retour est -1.

4.4 Tests

4.4.1 Nouvelle hiérarchie des tests/numéros des tests

Pour MedPlaus 6, la hiérarchie des tests a été retravaillée dans son entier. L'idée de base étant que les messages pertinents pour le codage soient affichés en premiers, suivis de ceux des tests sur les données administratives, puis, en dernier, ceux des problèmes structuraux (qui concernent avant tout l'IT et les fabricants de logiciels).

Avec cette révision, il existe une logique de test rigoureuse pour les années à venir.

Les tests de MedPlaus ont, de ce fait, reçu de nouveaux numéros.

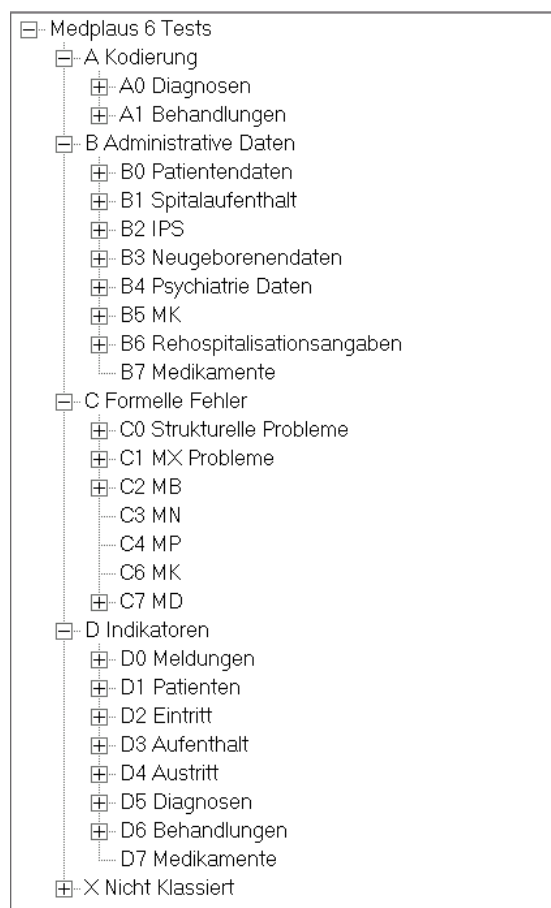


FIGURE 4.1 – La nouvelle hierarchie des tests

4.4.2 Concept de plausibilisation

Le concept de plausibilisation décrit les tests intégrés dans MedPlaus 6.

Le concept correspondant à cette version paraîtra après l'achèvement de la version définitive de MedPlaus 6 stand alone. Vous le trouverez sous :

- <http://www.freudiger.com/f/medplaus/> (Page d'accueil en langue française)
- <http://www.freudiger.com/medplaus/> (Page d'accueil en langue allemande)

4.5 Description des fonctions

4.5.1 Encodage des jeux de caractères

Ont cours : ASCII/ANSI (latin1). UTF n'est pas soutenu et, au vu des caractères transmis, n'est pas non plus une nécessité.

4.5.2 Description des types

Delphi	C	Description
Integer	int	Integer 32/64 bit
PChar	* char	Indicateur 32/64 bit pointant sur une chaîne de 8 bit de long et terminée par un zéro

4.5.3 fonction **MP_Create(MPDir : PChar; cLanguage : Integer) : Integer ;**

Initialise le système de test : Crée une instance et charge tous les tableaux des ressources nécessaires dans la mémoire vive.

Paramètre	Description
MPDir	Le répertoire dans lequel se trouvent les fichiers txt et ix5
cLanguage	0 : allemand 1 : français
Valeur en retour	<=0 : Erreur >0 : Descripteur de la fenêtre de test (réutilisé en tant que HDL dans les autres fonctions)

4.5.4 fonction **MP_setFormatString(formatString : PChar)**

Nouveau à partir de MedPlaus 6 : Définit le format de sortie des messages.

Si la fonction n'est pas employée, le format de sortie analogue à MedPlaus 5 est utilisé par défaut, .

Paramètre	Description
formatString	Chaîne qui définit le format de sortie des messages. Les champs à disposition sont adressés par le biais de {{Domaine.Champ}}. Là entre, il peut se trouver, outre '{{', de quelconques caractères.
Exemples	{{MB.1}} renvoie le champ 0.1.V01 de l'enregistrement MB ('MB') {{MB.4}} renvoie le champ 0.1.V04 de l'enregistrement MB (par ex. 'BE') {{SYS. Version}} renvoie la version du programme (par ex. '6.0.0.0 x64')
Domaines	MB - Tous les champs du MB sont, selon la spécification MS de l'OFS, appelables par leur rang MD - Tous les champs du MD sont, selon la spécification MS de l'OFS, appelables par leur rang SYS - Tous les messages système de MedPlaus sont appelables par le biais des champs qui suivent
Champs de domaine SYS	<i>StatYear</i> : Année statistique <i>Version</i> : Version de MedPlaus <i>LineNr</i> : Numéro de la ligne <i>ErrorNr</i> : Numéro long de l'erreur (> 5 positions pour les diagnostics, les traitements) <i>ErrorNrS</i> : Numéro court de l'erreur (5 positions. Sans le numéro du diagnostic, du traitement) <i>ErrorNrOld</i> : Numéro long de l'erreur, selon le système MP5. (Il peut y avoir un chevauchement dans le cas de nouveaux tests n'ayant pas d'ancien numéro d'erreur) <i>MsgType</i> : Erreur, avertissement, indication, ... <i>BUR</i> : Numéro REE de l'établissement <i>FID</i> : Numéro d'identification du cas <i>HKST</i> : Centre de charge OFS <i>KT</i> : Canton <i>Text0</i> : Texte bref (par ex. sans '1er diagnostic supplémentaire :') <i>Text1</i> : Texte long, comme précédemment (par ex. 'n diagnostic supplémentaire :') <i>Text2</i> : abrégé : '1er DS :' au lieu de '1er diagnostic supplémentaire :' <i>InvalidData</i> : Valeur causant le message (par ex. un code non valable)
Valeur en retour	0 : Fonction exécutée avec succès

4.5.5 function MP_TestRec(HDL : Integer ; Data : PChar ; ShowWindow : Integer ; MessageTypes : PChar ; cLanguage : Integer = -1) : Integer ;

Test d'un enregistrement MB avec ou sans enregistrement(s) supplémentaire(s).

Paramètre	Description
HDL	Descripteur de la fenêtre de test
Data	Pchar/*char-pointer sur les données : Les enregistrements MB, MN, MP, MD et MK peuvent être transférés, selon cet ordre, dans Data. Les enregistrements sont organisés selon la définition de l'interface concoctée par l'OFS. Des enregistrements différents sont séparés les uns des autres par CR, LF ou CR + LF. Les lignes vides et les enregistrements inconnus sont ignorés lors de ce processus.
ShowWindow	0 : Ne pas afficher la fenêtre des messages de MedPlaus 1 : Afficher la fenêtre des messages de MedPlaus,
MessageTypes	Obsolète. Un éventuel paramètre transmis est ignoré.
Valeur en retour	1 ou >1 : Nombre d'erreurs, avertissements, indications, erreurs structurelles (indépendamment des paramètres configurés dans MessageTypes) 0 : Sans erreur. Pas d'erreur structurelle ni d'avertissement. Pas d'indications non plus -1 : Le descripteur transmis n'est pas valable.
cLanguage	0 : allemand 1 : français

4.5.6 function MP_GetMessageList(HDL : Integer ; var ErrCount : Integer ; var WarnCount : Integer ; var HintCount : Integer ; var ControlCount : Integer ; var StructCount : Integer ; ErrList : PChar) : Integer ;

Donne le nombre et la liste des messages provenant de MP_TestRec().

Paramètre	Description
HDL	Descripteur de la fenêtre de test
ErrCount	Nombre d'erreurs dans l'enregistrement
WarnCount	Nombre d'avertissements dans l'enregistrement
HintCount	Nombre d'indications dans l'enregistrement
ControlCount	Valeur 0, non pertinente
StructCount	Valeur 0, non pertinente
ErrList	Retourne tous les messages d'erreurs. Chacun des messages est séparé des autres par CR + LF. ErrList (ErrCount + Warncount) doit mettre * 256 octets à disposition. C.f. Valeur en retour de TestRec.
Valeur en retour	<0 : Erreur 0 : Fonction conduite avec succès -1 : Descripteur non valable

4.5.7 function MP_Destroy(HDL : Integer) : Integer ;

Déblocage de l'instance/des ressources de MedPlaus.

Paramètre	Description
HDL	Descripteur de l'instance de MedPlaus
Valeur en retour	0 : Fonction conduite avec succès -1 : Descripteur non valable

4.5.8 function MP_VersionAppName() : PChar ;

Nom de l'application

Paramètre	Description
Valeur en retour	Nom de l'application

4.5.9 function MP_VersionNumber() : PChar ;

Numéro de version de l'application

Paramètre	Description
Valeur en retour	Numéro de version de l'application

4.5.10 function MP_VersionDate() : PChar ;

Date de l'application

Paramètre	Description
Valeur en retour	Date de l'application

4.6 MP6 DLL-DynTest

Le fichier mp6DllDynTest.exe³ démarre une surface graphique permettant de tester les appels de la DLL (lien dynamique) grâce à un enregistrement faisant office d'exemple.

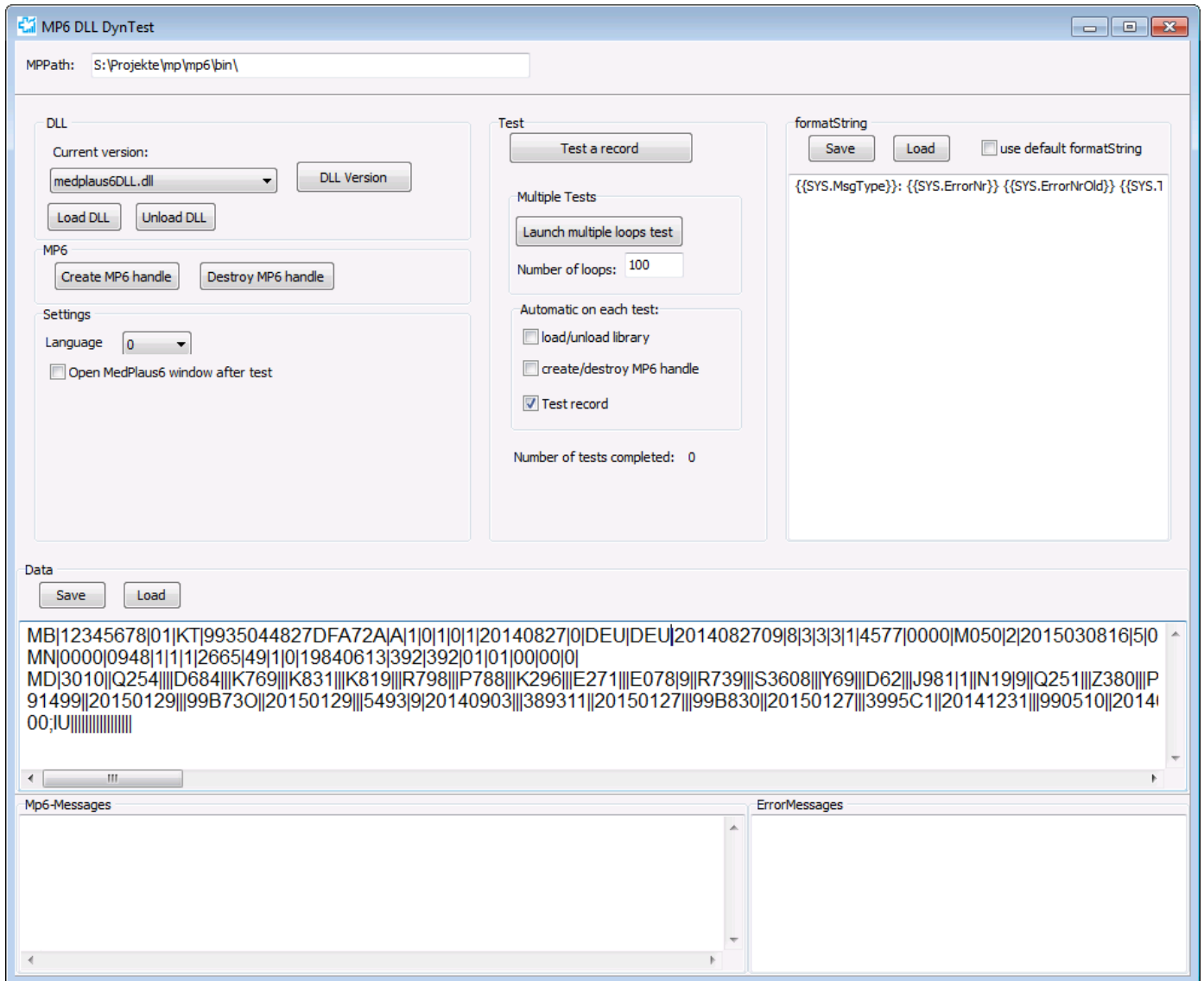


FIGURE 4.2 – MP6 DLL-DynTest

3. Alternative : mp6DLLDynTest64.exe pour tester la DLL 64 bit de MedPlaus

4.6.1 Préparations du test

La procédure de base est la suivante :

1. Indiquez lecteur et chemin où se trouvent les fichiers de la DLL de MedPlaus.
Il est indispensable de terminer le chemin par une barre oblique inversée (par ex. `c:\programme\mp6070\`)



FIGURE 4.3 – MP6 DLL-DynTest : Chemin

2. Choisissez ensuite la DLL de MedPlaus désirée :
 - Par défaut : `medplaus6DLL.dll`
 - Pour la recherche des erreurs : `medplaus6DLL.BigLog.dll`
3. Chargez la DLL par un clic sur *Load DLL*

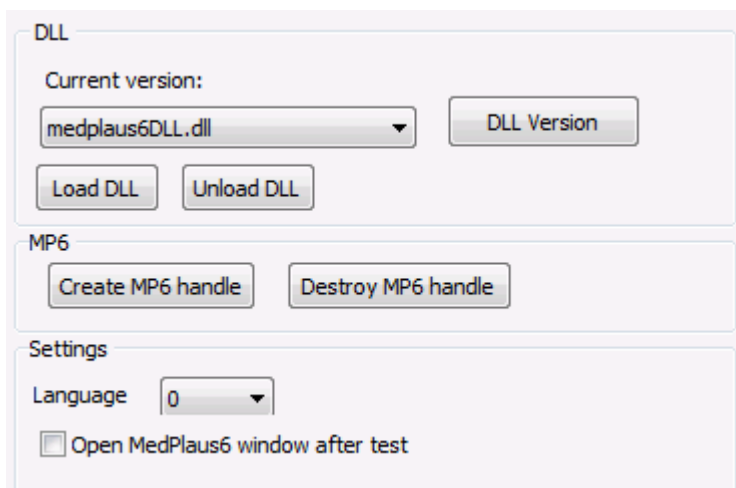


FIGURE 4.4 – MP6 DLL-DynTest : Load DLL

4. Si besoin est, faites s'afficher d'autres informations concernant la DLL, comme la version et la date (par un clic sur *DLL Version*)
5. Sélectionnez les paramètres désirés sous 'Settings' :
 - a) La langue dans le menu déroulant *Language*
 - 0 : Allemand
 - 1 : Français
 - b) Lorsque l'option *Open MedPlaus6 window after test* est activée, l'enregistrement et les messages sont affichés dans une fenêtre MedPlaus⁴ séparée.
6. Créez un descripteur unique avec *Create MP6 handle*

4. mp6show.exe

4.6.2 Conduite du test

Section Test

Par le biais de *Test a record*, un test est conduit une seule fois.

- L'enregistrement de test se trouve dans la zone 'Data'
- Les résultats sont affichés, soit dans la zone 'Messages', soit dans une fenêtre MedPlaus séparée

Dans la zone 'Multiple Tests', un test peut être effectué aussi souvent que désiré. La valeur de 'Number of loops' détermine le nombre d'exécutions.

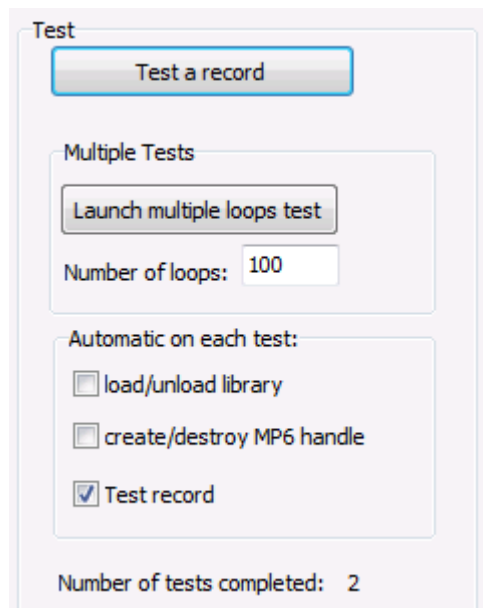


FIGURE 4.5 – MP6 DLL-DynTest : Test unique/multiple

Avec *Automatic on each test*, les actions suivantes peuvent être exécutées automatiquement :

- Charger/Enlever une DLL (*load/unload library*)
- Créer/Détruire un descripteur (*create/destroy MP6 handle*)
- Test d'un enregistrement (*Test record*)

La multiple exécution du test le ralentit, étant donné que la DLL doit à chaque fois être de nouveau chargée.

En dessous des paramètres du test, le nombre de tests complets est encore affiché (*number of tests completed* :).

Section Data

L'enregistrement (avec enregistrements supplémentaires) à tester se trouve dans la section 'Data'

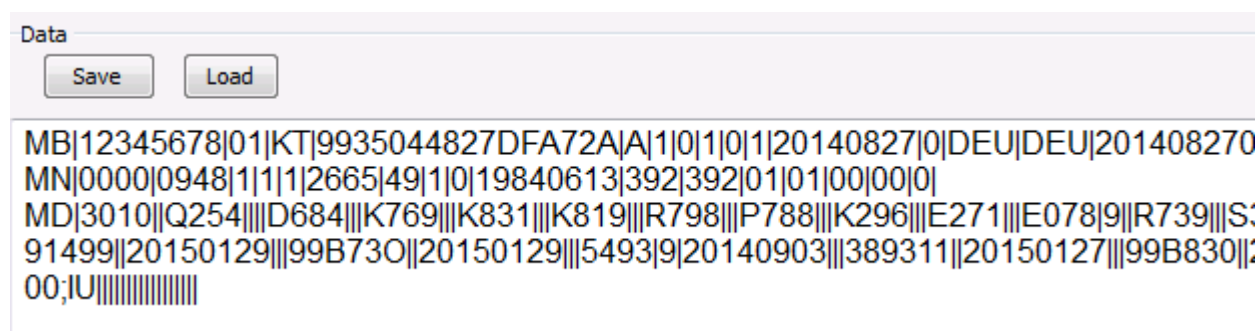


FIGURE 4.6 – MP6 DLL-DynTest : Section Data

Avec *Save*, l'enregistrement est sauvegardé comme fichier texte (mp6DllDynTestData.txt).

Avec *Load*, le contenu du mp6DllDynTestData.txt est lu et affiché.

Ces fonctions sont utiles, lorsque vous entreprenez de faire de nombreuses modifications à l'enregistrement test à contrôler. Au lieu de les faire dans l'interface graphique, vous pouvez les effectuer dans un éditeur de texte.

Section Messages : Messages

La zone gauche de la section Messages affiche les messages MedPlaus 6 (erreurs, avertissements et indications) résultant du test de l'enregistrement.

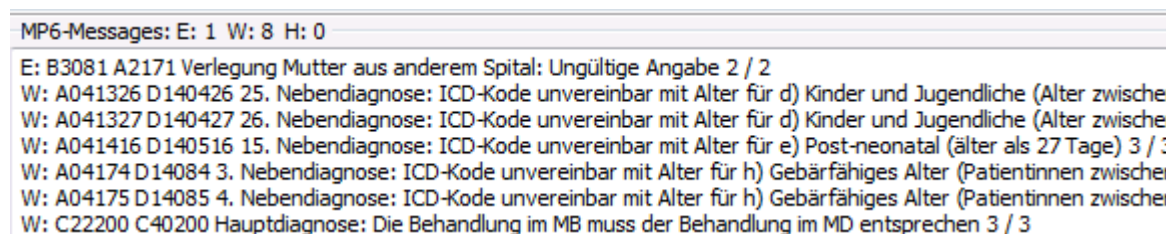


FIGURE 4.7 – MP6 DLL-DynTest : Messages MP6

Veuillez prêter attention aux éléments suivants :

- Dans la partie supérieure sont listés les nombres d'erreurs (E), d'avertissements (W) et d'indications (H) affichés
- Les messages ne sont pas triés
- Le format de sortie correspond au standard de MedPlaus (par défaut) ou au format personnalisé (MP_setFormatString())

Section Messages : Messages d'erreurs

La partie droite de la section Messages affiche les messages d'erreur du programme.

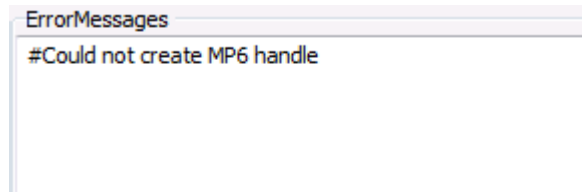


FIGURE 4.8 – MP6 DLL-DynTest : Messages d'erreur

4.6.3 Informations supplémentaires

Changement de la DLL

Pour changer la DLL, vous devez d'abord ôter l'ancienne. La procédure est la suivante :

1. Cliquez sur *Destroy MP6 handle* pour le détruire
2. Enlevez la DLL par un clic sur *Unload DLL*
3. Chargez la nouvelle DLL (c.f. Préparations du test)

Lorsque les points *load/unload library* et *create/destroy MP6 handle* sont activés dans la zone *Automatic on each test*, les premières deux étapes sont sautées. L'on peut sélectionner directement une nouvelle DLL et procéder à des tests avec elle.

Affichage des données dans la fenêtre de MedPlaus

Lorsqu'*Open MedPlaus6 window after test* est activée, l'enregistrement testé et les résultats sont affichés dans une fenêtre séparée. Dans la partie supérieure, s'affichent les contenus, tandis qu'au niveau inférieur sont listés les erreurs, avertissements et indications trouvés.

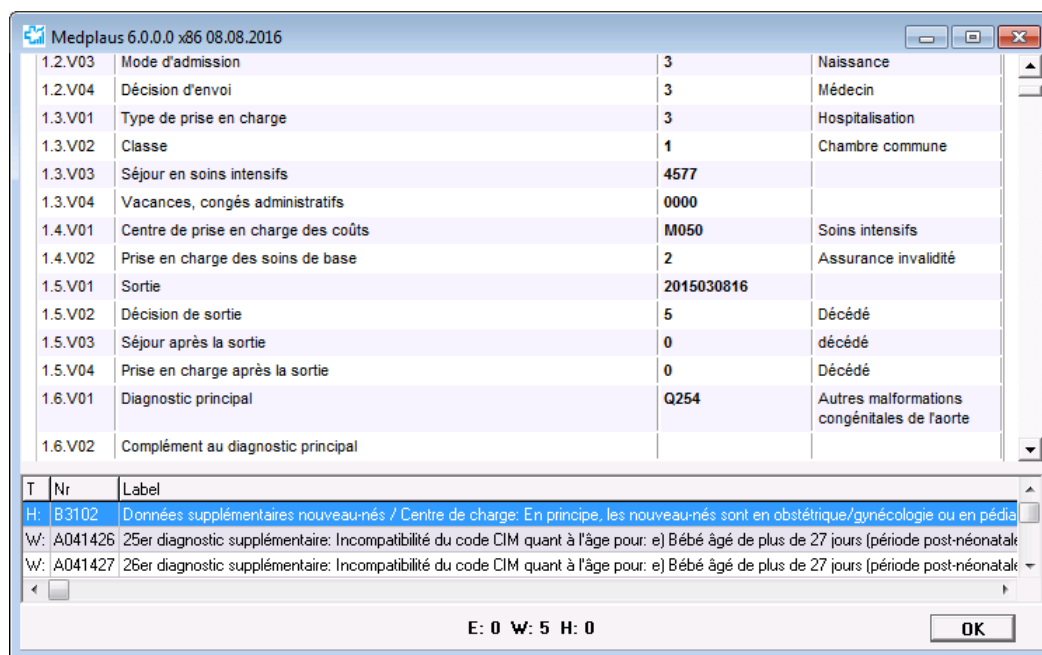


FIGURE 4.9 – mp6show.exe

La barre de titre vous informe sur la version et la date de la DLL de MedPlaus.

La barre d'état liste le nombre des erreurs (E), avertissements (F) et indications (H) affichés.

Le bouton *OK* ferme la fenêtre. Après quoi, vous vous retrouvez sur la surface de la MP6 DLL DynTest (resp. MP6 DLL DynTest 64-bit).

Section formatString

Le formatString est un nouveau format introduit avec MedPlaus 6. Il permet la création personnalisé de chaînes de caractères avec constantes et variables.

Vous trouverez des informations détaillées quant au format dans la section 'fonction MP_setFormatString()'.

Utilisation

- Formatage personnalisé de la sortie des messages dans la zone MP6-Messages de la Gui de test
- Formatage personnalisé pour utilisation avec fonction MP_GetMessageList()

Avec formatString, des statements SQL peuvent, par exemple, être générés sans grand embarras. Celles-ci permettent la sauvegarde et l'évaluation des messages de MedPlaus dans une banque de données.

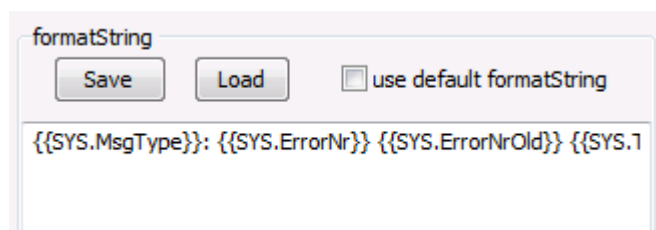


FIGURE 4.10 – MP6 DLL-DynTest : formatString

Avec *Save*, le formatstring est sauvegardé en tant que fichier texte (mp6DllDynTestFormatString.txt). Avec *Load*, le contenu du mp6DllDynTestFormatString.txt est lu et affiché.

Ces fonctions sont utiles, si vous désirez entreprendre de nombreux changements au formatstring personnalisé. Vous pouvez effectuer celles-ci dans un éditeur de texte plutôt que dans la Gui de test.

Lorsque *use default formatString* est activée, c'est le standard MedPlaus 6 qui est utilisé. Il correspond à celui de MedPlaus 5.

4.7 Recherche d'erreurs

La MedPlaus DLL est livrée avec des versions DLL rapides et productives en 32 et 64 bit. Elles n'établissent aucun fichier LOG par défaut. Dans le cas où vous rencontreriez des problèmes lors de l'intégration de la DLL, procédez de la manière suivante :

1. Sécurisez la DLL productive (medplaus6DLL.dll ou medplaus6DLL64bit.dll).
2. Utilisez, au lieu de la MedPlaus 6 DLL productive, la version du répertoire \DLL_BigLog. Cette version offre une fonctionnalité LOG détaillée qui vous aidera à cerner le problème.
 - a) Copiez la DLL BigLog⁵ dans le répertoire principal et renommez-la medplau6DLL.dll.
 - b) Communiquez avec la DLL.
 - c) Analysez ensuite le fichier medplaus6DLL_debug.txt placé dans le répertoire c :\temp⁶.
Ce problème résolu, vous pourrez à nouveau copier la version productive de la MedPlaus 6 DLL dans le répertoire principal.
3. Si la source du problème reste obscure, veuillez s.v.p. procéder de la façon suivante :
 - a) Supprimez le contenu du fichier medplaus6DLL_debug.txt du répertoire c :\temp
 - b) Communiquez de nouveau avec la DLL BigLog
 - c) Envoyez-nous, par courriel, le contenu complet du fichier LOG à medplaus@freudiger.com

5. Procédure analogue en ce qui concerne la version 64 bit, en adaptant les noms de fichiers

6. c :\temp doit déjà exister pour que le fichier Debug-Log y soit écrit

5 Divers

5.1 Support par le fabricant

D'autres questions ? Profitez du support de la Freudiger EDV-Beratung. Envoyez une description exacte du problème (év. avec extraits du code source et printscreens anonymisés), y compris l'information quant aux versions utilisées, à medplaus@freudiger.com.

Si vous désirez transmettre des données sensibles, vous pouvez procéder par le biais de notre portail BESTAT. A ce sujet, vous voudrez bien contacter notre hotline (031 318 17 24).

5.2 Historique du document

18.01.2023 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.7.0)
19.07.2022 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.6.0)
04.01.2021 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.5.0)
28.04.2020 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.4.2)
04.03.2020 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.4.1)
13.12.2019 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.4.0)
09.07.2019 :	Actualisation de la documentation (état : MedPlaus DLL 6.0.3.1)

5.3 Version

Version : 6.070
Dernière modification : 18.01.2023
Statut : Public