



Projet IGH DataExpert

Spécifications

Conditions

Date : 08.04.10

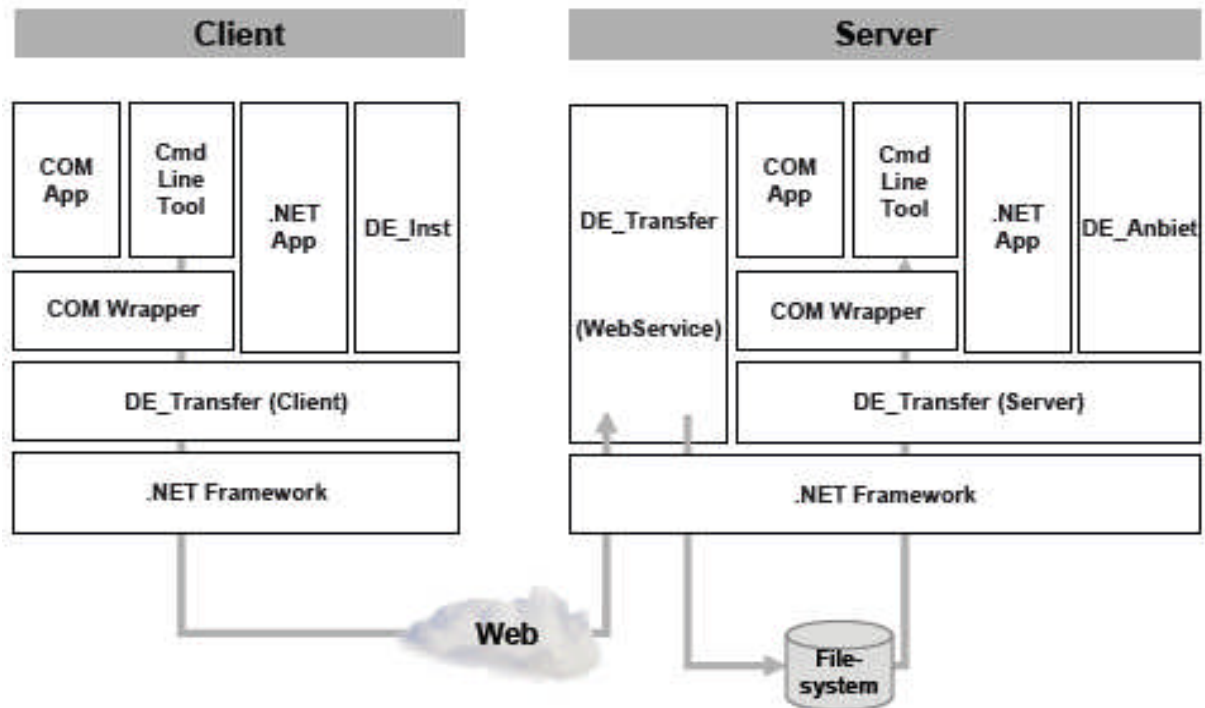
Version : 1.5.0.0

1 Introduction

Ce document présente les spécifications nécessaires à la réalisation du projet DataExpert.

Nous entendons sous fournisseur ou Server le distributeur ou fabricant auquel commandes ou demandes sont adressées. Sous Client ou installateur, nous entendons l'acheteur ou le demandeur d'offre.

2 Structure



Structure des composantes

Remarques:

- DataExpert se compose de trois modules principaux
 - DE_Transfer : communication
 - DE_Inst : configuration Client
 - DE_Anbiet : configuration Server
- DE_Transfer se compose également de trois modules
 - Client : communication entre Client et WebService, tenue des données de configuration
 - WebService : communication avec Client
 - Server : accès aux données Client, tenue des données de configuration
- Les applications COM accèdent à DE_Transfer au moyen d'un COM Wrapper
- Les applications NET peuvent accéder à DE_Transfer sans COM Wrapper
- Les outils de configuration DE_Inst et DE_Anbiet accèdent par DE_Transfer aux données de configuration (Exemple d'accès d'une application NET à DE_Transfer)
- L'outil Command-line accède par COM Wrapper à DE_Transfer (Exemple d'accès d'une application COM à DE_Transfer)

2.1 Principes de structure

La structure de DataExpert est basée sur les principes suivants:

1. **Facteur:** DataExpert est uniquement responsable de la transmission et n'intervient d'aucune manière dans le contenu des données, à la manière du facteur s'occupant du transport et de la remise du courrier sans en connaître le contenu.
2. **Pas de Server central:** DataExpert ne connaît ni Server central, ni poste Clearing. Client et fournisseurs communiquent directement et indépendamment avec des partenaires DataExpert. Un monopole d'information ne doit fondamentalement pas se former (comme par exemple sur des sites commerciaux).
3. **Invisible:** DataExpert met à disposition d'applications nouvelles ou existantes une fonctionnalité de communication B2B. DataExpert ne communique pas, en soi, directement avec les utilisateurs. Les outils basiques de configuration et de test constituent la seule dérogation à cette règle.
4. **Principe Pull:** le Client est responsable de l'établissement de la communication, c'est-à-dire que le Client s'annonce auprès du fournisseur et échange avec celui-ci les informations qu'il juge opportunes. Le fournisseur ne s'annonce jamais auprès du Client.
5. **Authentification avec Windows Logon:** DataExpert utilise à nouveau l'infrastructure de sécurité existante de Windows. Toutes les authentifications s'effectuent du côté du Client par Windows Logon. Divers droits peuvent être accordés selon Windows Logon (autorisation). Du côté du fournisseur, seuls les "trusted entities" ont un accès direct à l'ordinateur DataExpert ou aux composantes DataExpert. Aucune authentification ou autorisation spéciale n'est effectuée.
6. **Libre de base de données:** Aucune banque de données n'est requise, tant de la part du fournisseur que de celle du Client. L'engagement et le choix d'une éventuelle banque de données est décidée du côté de l'installateur par le fournisseur de son logiciel d'application et du côté du fournisseur par son département IT respectif.
7. **Cryptage:** Les données destinées à être transmises par Internet sont cryptées. Les données non-destinées à la transmission sont protégées en dehors de DataExpert par l'infrastructure du système d'exploitation.

3 Spécification détaillée des composantes

3.1 DE_Transfer Client

3.1.1 Méthodes (avec Webservice)

3.1.1.1 GetBizMessageList

Interface COM:

```
HResult GetBizMessageList([in]string partnerID, [out]string listXML, [retval,out]transactionID)
```

Interface Webservice:

```
public string GetBizMessageList(string username, string password, out string list)
```

Fonctionnalité:

- Crée une liste XML de tous les BizMsgs à disposition de ce User pour un Download.

La liste contient par BizMessage:

- ID (=Filename)
- Longueur (en Bytes)
- Date de création / modification

Remarques:

- La liste XML est indiquée, du côté du Server, par tous les files se trouvant dans la liste Download avec le préfixe "username".
- La liste correspond au schéma XML 'GetList.xsd' de la table de configuration.

3.1.1.2 ReadBizMessage

Interface COM:

HResult ReadBizMessage([in]string partner, [in] bizMsgID, [out]string bizMsg, [out]byte[] attachmentContent, [out]string attachmentFileExt, [retval,out]string transactionID)

Interface WebService:

public string ReadBizMessage2 (string username, string password, string bizMsgID, out string bizMsg, out Attachment attachment)

Fonctionnalité:

- Validation du droit d'accès sur BizMsg: BizMsgID présente le préfixe 'username'.
- Fournit dans le paramètre 'bizMsg' le contenu des Files. En cas d'apparition d'un Attachment, celui-ci est retourné à la structure Attachment. Le genre Attachment est une structure de données contenant les Bytes de l'Attachment de même que le File-Extension de l'Attachment. En cas de Message sans Attachment, ce paramètre sera Null, respectivement Nothing.

3.1.1.3 ReadIkkBizMessage

Interface COM:

HResult ReadIkkBizMessage([in]string partner, [out]string ikkBizMsg)

Interface WebService:

public string ReadIkkBizMessage (string username, string password, out string ikkbizMsg)

Fonctionnalité:

- Fournit dans le paramètre 'ikkBizMsg' le contenu du File ikk.
- Un String vide sera retourné en cas de manque de File ikk du côté du Server pour l'utilisateur.

3.1.1.4 RemoveBizMessage

Interface COM:

HResult RemoveBizMessage([in]string partnerID, [in]string BizMsgID, [retval,out] transactionID)

Interface WebService:

public string RemoveBizMessage (string username, string password, string bizMsgID)

Fonctionnalité:

- Validation du droit d'accès sur BizMsg: BizMsgID présente le préfixe 'username'.
- Supprime le File 'BizMsgID' de la liste Download.

3.1.1.5 WriteBizMessage

Interface COM:

HResult WriteBizMessage([in]string partnerID, [in] string bizMsg, [in]byte[] attachmentContent, [in]string attachmentFileExt, [retval,out] transactionID)

Interface WebService:

public string WriteBizMessage2 (string username, string password, string bizMsg, Attachment attachment)

Fonctionnalité:

- Validation du contenu de 'bizMsg' avec le Parser XML.
- Ecrit le contenu de 'bizMsg' dans le Upload Directory.
- Le ID utilisé dans nom du File correspond à Transaction ID.
- En cas d'envoi d'un Attachment, celui-ci est fourni avec la structure Attachment. Le genre Attachment est une structure de données contenant les Bytes de l'Attachment de même que le File-Extension de l'Attachment. En cas d'envoi sans Attachment, ce paramètre sera Null, respectivement Nothing.

3.1.1.6 TriggerIkkBizMessage

Interface COM:

HResult TriggerIkkBizMessage([in]string partnerID)

Interface WebService:

public string TriggerIkkBizMessage(string username, string password)

Fonctionnalité:

- Ecrit une requête (fichier) pour l'actualisation du File ikk dans Uploadikk Directory.

3.1.1.7 GetCatalogList

Interface COM:

HResult GetCatalogList([in]string partnerID, [out]string listXml, [retval,out] transactionID)

Interface WebService:

public string GetCatalogList (string username, string password, out string listXml)

Fonctionnalité:

- Etabli une liste XML de toutes les données (catalogue) présentes dans la liste Catalog prêtes au Download.

3.1.1.8 ReadCatalog

Interface COM:

HResult ReadCatalog([in]string partner, [in] catalogID, [out]string catalogXml, [retval,out] string transactionID)

Interface WebService:

public string ReadCatalog (string username, string password, string catalogID, int offset, int size, out byte[] catalogFragment, out long length)

Fonctionnalité:

- Fournit dans le paramètre 'catalogFragment' le contenu du File.

Remarque:

- Lors du Download du catalogue, un Event est déclenché à chaque chargement d'un fragment de catalogue. L'Event indique l'état d'avancement du Download. Les indications sont fournies en pourcent, c'est-à-dire avec une valeur entre 0 et 100.

3.1.1.9 ReadSchema

Interface COM:

HResult ReadSchema([in]string partner, [uint] branchenID, [uint] processID, [out]string schema, [retval,out]string transactionID)

Interface WebService:

```
public string ReadSchema2 (string username, string password, uint branchenID, uint processID, out string schema)
```

Fonctionnalité:

- Fournit dans le paramètre 'schema' le contenu du File "BizMessage_'BranchenID'_'ProcessID'_'PartnerID'.xsd" à partir de la liste de configuration du Server.

Remarque:

- 'PartnerID' correspond à 'PartnerID' du fournisseur.

3.1.1.10 ChangePassword

Interface COM:

```
HResult ChangePassword([in]string PartnerID, string newPassword, [retval,out]transactionID)
```

Interface Webservice:

```
public string ChangePassword(string username, string password, string newPassword)
```

Fonctionnalité:

- Change dans le File de configuration "ClientPartner.xml" le mot de passe associé au PartnerID transmis en 'newPassword'.
- Change dans le File de configuration "ServerUser.xml" le mot de passe de 'username' en 'newPassword'.

3.1.2 Méthodes (sans Webservice)

3.1.2.1 ReadPartner

Interface COM:

```
HResult ReadPartner( [out]string partnerXml)
```

Fonctionnalité générale / Fonctionnalité Client:

- Fournit dans 'partnerXml' les données ClientPartner comme XML, donc le contenu du File 'ClientPartner.xml' de la liste de configuration.

Remarque:

- DE_Inst s'ingère avec cette fonction dans les données Partner.

3.1.2.2 WritePartner

Interface COM:

```
HResult WritePartner( [in]string partnerXml)
```

Fonctionnalité générale / Fonctionnalité Client:

- Valide le contenu de 'partnerXml' en accord avec le schéma 'ClientPartner' dans la liste de configuration.
- Vérifie la précision de tous les 'PartnerID'.
- Enregistre le contenu de 'partnerXml' dans le File 'ClientPartner.xml' dans la liste de configuration.

Remarque:

- DE_Inst s'ingère avec cette fonction dans les données Partner.

3.1.2.3 ReadUser

Interface COM:

HResult ReadUser([out]string userXml)

Fonctionnalité générale / Fonctionnalité Client:

- Fournit dans 'userXml' les données ClientUser comme XML, donc le contenu du File 'ClientUser.xml' de la liste de configuration.

Remarque:

- DE_Inst s'ingère avec cette fonction dans les données Partner.

3.1.2.4 WriteUser

Interface COM:

HResult WriteUser([in]string userXml)

Fonctionnalité générale / Fonctionnalité Client:

- Valide le contenu de 'userXml' en accord avec le schéma 'ClientUser' dans la liste de configuration.
- Vérifie la précision de tous les 'username'.
- Enregistre le contenu de 'userXml' dans le File 'ClientUser.xml' dans la liste de configuration.

Remarque:

- DE_Inst s'ingère avec cette fonction dans les données Partner.

3.1.2.5 GetProperty

Interface COM:

HResult GetProperty([in]string name, [out]string value)

Fonctionnalité:

- Identique à GetProperty du côté du Server

Propriétés soutenues:

- 'LogFile' : Filename du Log-File (chemin d'accès absolu inclus)
- 'ConfigDir' : Chemin d'accès à la liste de configuration
- 'BlockSize' : Blocksize en Bytes du Download du catalogue

3.1.2.6 SetProperty

Interface COM:

HResult SetProperty([in]string name, [in]string value)

Fonctionnalité:

- met la Property spécifiée.

Propriétés soutenues:

- 'BlockSize' : Blocksize en Bytes du Download du catalogue

Remarque:

- Les valeurs Property sont tenues de manière persistante.

3.1.3 Events

3.1.3.1 ReadCatalogProgress

Interface COM:

ReadCatalogProgress([in]int progress)

Fonctionnalité:

- Lors du Download du catalogue (ReadCatalog) cet Event est déclenché à chaque chargement d'un fragment de catalogue. Le paramètre "progress" indique l'état d'avancement du Download. Les indications sont fournies en pourcent, c'est-à-dire avec une valeur entre 0 et 100.

4 DE_Inst

DE_Inst est la surface graphique servant à la mutation des données de configuration XML "ClientUser.xml" et "ClientPartner.xml" du côté du Client. La mutation s'effectue au moyen d'un DB Grid.

DE_Inst présente les fonctionnalités suivantes:

- Saisir, modifier, supprimer, enregistrer
 - toutes les données ClientUser
 - toutes les données ClientPartner
- Modifier les mots de passe Partner
 - modifie le mot de passe localement et sur le Server du partenaire correspondant (appel de ChangePassword)
 - Les données préalablement modifiées sont enregistrées ou rejetées avant chaque changement de mot de passe
- Importation de données ClientPartner / ClientUser
 - Validation des données au moyen du schéma correspondant dans la liste de configuration
 - Les données importées s'ajoutent aux données existantes ou les remplacent
- Exportation de données ClientPartner / ClientUser (correspond à la fonctionnalité "save as")
- Lors d'un appel de DE_Inst à l'aide d'un argument, celui-ci est interprété comme le Filename d'un fichier ClientPartner à importer, c'est-à-dire que les données seront importées
- Après l'exécution de modifications, un dialogue "Enregistrer oui/non" apparaît avant de quitter DE_Inst
- Mise en place de la configuration Proxy

Remarques:

- DE_Inst s'ingère dans les données ClientUser et ClientPartner par le biais de l'interface DE_TransferClient
- Les données importées dans DE_Inst sont produites de manière typique par un fournisseur à l'aide de DE_Anbiet

5 DE_Anbiet

DE_Anbiet est la surface graphique servant à la mutation du côté du Server pour:

- les données de configuration "ServerUser.xml"
- le schéma "BizMessage_'BranchenID'_'ProcessID'_'PartnetID'.xsd".

La mutation s'effectue au moyen d'un DB Grid.

DE_Anbiet présente les fonctionnalités suivantes:

- Saisir, modifier, supprimer, enregistrer
 - toutes les données ServerUser
- Importation de données ServerUser
 - Validation des données au moyen du schéma "ServerUser" dans la liste de configuration
 - Les données importées s'ajoutent aux données existantes ou les remplacent
- Exportation de données ServerUser (correspond à la fonctionnalité "save as")
- Paramètres pour un User-Export d'un fichier ClientPartner avec les informations de communication pour ce Server (WebService URL, etc.).
- Chargement
 - du schéma BizMessage original "BizMessage.xsd"
 - du schéma BizMessage actuel "BizMessage_'PartnerID'.xsd" (Default)
- Modification de la valeur "optional" ou "required" de l'attribut "use" du schéma "BizMessage" chargé
- Enregistrement du schéma BizMessage sous "BizMessage_'PartnerID'.xsd"

Remarque:

- Il est donné un schéma BizMessage par fournisseur. Aucun "type extensions" etc n'est utilisé.

6 Divers

6.1 Structure de l'arborescence

Toutes les composantes et données spécifiques à DataExpert se trouvent dans une arborescence présentant la structure suivante:

- IGH
 - DataExpert
 - **Client** : contient les composantes exécutables du logiciel Client
 - Log: : contient les Log-Files
 - DE_ClientLog.csv
 - Config: : contient les Files de configuration et les schémas spécifiques à DataExpert (p.ex. pour ClientUser, ClientPartner)
 - ClientUser.xml
 - ClientPartner.xml
 - ClientUser.xsd
 - ClientPartner.xsd
 - Catalog.xsd (schéma pour catalogue)
 - GetList.xsd (schéma pour GetBizMessageList)
 - **Server** : contient les composantes exécutables du logiciel Server
 - Upload : contient les BizMsgs transmis par le Client
 - Download : contient les BizMsgs à transmettre au Client
 - Uploaddikk : contient les demandes d'actualisation transmises par le Client
 - Downloadikk : contient le fichier actuel ikk (pour le Client)
 - Catalog : contient les données du catalogue
 - Log : contient le Log-File
 - DE_ServerLog.csv
 - Config : contient les Files de configuration et les schémas spécifiques à DataExpert (p.ex. pour ServerUser, BizMessages et catalogues)
 - ServerUser.xml
 - ServerUser.xsd
 - BizMessage.xsd (le schéma BizMessage original)
 - BizMessage_'BranchenID'_'ProcessID'_'PartnerID'.xsd (le schéma BizMessage pour ce fournisseur)
 - Catalog.xsd

- GetList.xsd (schéma pour GetBizMessageList etc.)

6.2 Données d'autorisation et de relation

Toutes les données de configuration sont contenues en .XML Files dans Config Directory.

6.2.1 Tableau "ClientUser"

Est utilisé pour l'autorisation d'un appel de DE_Transfer du côté Client

WindowsLogon	: String / PrimaryKey / WindowsLogon nom de l'utilisateur
WriteBizMsg	: Boolean / Autorisation des méthodes <ul style="list-style-type: none"> • WriteBizMsg
ReadBizMsg	: Boolean / Autorisation des méthodes <ul style="list-style-type: none"> • GetBizMsgList • ReadBizMsg • RemoveBizMsg
Catalog	: Boolean / Autorisation des méthodes <ul style="list-style-type: none"> • GetCatalogList • ReadCatalog
Administration	: Boolean / Autorisation des méthodes <ul style="list-style-type: none"> • ChangePassword • readUser • writeUser • readPartner • writePartner

Remarques:

- Les méthodes non-produites telles que
 - readSchema
 - getProperty
 sont autorisées pour tous les utilisateurs avec une inscription dans le tableau ClientUser.
- Les utilisateurs non-produits n'ont aucune autorisation d'appel de méthodes.
- Une seule inscription par Windows Logon est possible.

6.2.2 Tableau "ClientPartner"

Contient les informations de liaison pour l'appel d'un WebServices du côté Client.

PartnerID	: String / Primary Key
Name	: String / Nom du partenaire (p.ex. "Sanitas Troesch")
WebServiceURL	: String / URL de DataExpert WebServices
Username	: String / Nom d'utilisateur du client auprès de ce fournisseur
Password	: String / Mot de passe du client auprès de ce fournisseur

Remarque:

- La mention PartnerID est un numéro d'identification explicite du fournisseur valable dans toute la Suisse (p.ex. "1910" = Georg Fischer Rohrleitungssysteme (Schweiz) AG).

6.2.3 Tableau "ServerUser"

Est utilisé pour l'autorisation, du côté Server, d'un appel de WebServices par un Client.

Username	: String / PrimaryKey / Identification du client
Password	: String / Mot de passe de ce client
WriteBizMsg	: Boolean / Autorisation de la méthode Webservice "WriteBizMsg"
PID	: String / Paynet ID (optionnel)

Remarques:

- Toutes les méthodes, sauf "WriteBizMsg" sont autorisées pour tous les utilisateurs avec une inscription dans le tableau ClientUser. Les utilisateurs non-produits n'ont aucune autorisation d'appel de méthodes.
- De cette manière un appel de ReadBizMsg est implicitement permis à tous les clients.

6.3 Noms de Files

Les informations Meta inhérentes aux BizMsgs telles que expéditeur / destinataire ou les ID sont codifiées dans les noms de Files. Les files se trouvant dans les arborescences Upload et Download, du côté du Server, présentent la structure nominative suivante:

username_datum_zeit_id.xml

avec:

- username : nom de l'utilisateur à qui le message est destiné, ou de qui il émane
- datum : date d'élaboration dans le format JJMMTT
- zeit : heure d'élaboration dans le format hhmms
- id : TransactionID produit (GUID)

Fichiers ikk:

username_id.xml

- username : nom de l'utilisateur à qui le message est destiné, ou de qui il émane
- id : ikk (conditions client individuelles)

Demande d'actualisation ikk:

username_refresh.xml

- username : nom de l'utilisateur de qui émane la demande d'actualisation

(contenu de la demande: <ItemID="Client_refresh" Date="2010-03-17" />

6.4 Extension de nom de Files

DataExpert utilise les extensions de nom de Files suivants:

- .dep : données Partner à importer du côté du Client (produit de DE_Anbiet)
- .csv : fichiers Log
- .xml : tout autre fichier XML

6.5 Volumes de transport

- Les fichiers catalogue peuvent atteindre des grandeurs > 1 MByte
- Les schémas, listes et BizMsgs présentent des grandeurs KByte de l'ordre de un à deux chiffres
- Bande passante du côté du Client >= 28 Kbits/sec

6.6 Alimentation du catalogue

Les catalogues sont, lors de l'enregistrement dans la liste des catalogues, automatiquement remis à DataExpert.

Remarques:

- La validation du catalogue est du ressort de l'application enregistreuse
- Respecter la convention de nom selon le document "Codification-fichiers-xml.pdf"

6.7 Sécurité

- Authentification
 - Côté Client : par Windows Logon

- Webservice : par Username/Mot de passe selon tableau de configuration "ServerUser"
 - Côté Server : par Windows Logon
- Autorisation
 - Côté Client : par tableau de configuration "ClientUser"
 - Webservice : par tableau de configuration "ServerUser"
 - Côté Server : par Windows Administration
- Accès depuis Internet / Firewalls
 - selon concepts de sécurité spécifiques aux entreprises
 - http / HTTPS Ports ouverts de et vers Internet
- Transport
 - par HTTP / HTTPS
- Tenue des données
 - Côté Client : en dehors de DataExpert, par infrastructure système d'exploitation
 - Côté Server : en dehors de DataExpert, par infrastructure système d'exploitation

6.8 XML Parser

.NET XML Parser est utilisé par DataExpert pour toutes les validations XML.

6.9 Log File

Log-File contient les informations suivantes:

- Timbre horaire : dans le format TT.MM.JJ HH:MM:SS
- Utilisateur : Windows Logon (Client), Username (Server)
- Type : Entry, WS, Debug, Exit, Abort
- Méthode : nom de la méthode appelée (y c. URL avec WS)
- Argument : version abrégée des arguments transmis (Entry, WS)
- TransactionID : TransactionID (si disponible)
- ReturnValue : (Exit)
- Comment : texte, contient

Remarques:

- Le LogFile est écrit CSV (Excel) et non pas comme File XML. Ceci permet un ajout rapide d'inscription Log par le biais d'un File-Append.
- Server et Client utilise le même format Log File.
- Le volume du Log-File est illimité.
- Les Log-Files peuvent être examinés et analysés avec Excel.

6.10 Installation

Un programme permet l'installation des composantes du logiciel DataExpert (Setup).

Remarques:

- .NET Framework est installé indépendamment de DataExpert
- Les éléments suivants sont configurables au moment de l'installation:
 - Client
 - Emplacement de l'arborescence IGH
 - Server
 - Emplacement de l'arborescence IGH
 - PartnerID de ce fournisseur (pour tableau ClientPartner)
 - PartnerName de ce fournisseur (pour tableau ClientPartner)
 - URL du WebServer (pour tableau ClientPartner, pour Deployment du Webservice)
- Pendant l'installation côté Server:
 - le "BizMessage_*.xsd" d'origine sera copié sur BizMessage_*_*_PartnerID'.xsd spécifique au Partner

7 Système demandé

7.1 Hardware

Scenario	Required Processor	Recommended Processor	Required RAM	Recommended RAM
Client (Windows Forms and Windows Services)	Pentium 92 MHz*	Pentium 90 MHz or faster	32 MB*	96 MB or higher
Server	Pentium 133 MHz*	Pentium 133 MHz or faster	128 MB*	256 MB or higher

*or the minimum required by the operating system, whichever is higher.

Env. 35 MB sont nécessaires sur le disque pour l'installation de .NET Framework.

7.1 Software

Client:

Win98, WinMe, WinNT, Win2K, WinXP, Vista, Windows 7
.NET Framework 2.0

Server:

Win2K Server, Windows 2003 Server, Windows 2008 Server
IIS / .NET Framework2.0 / X.509