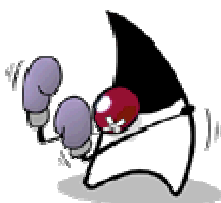


## Laboratoire de Technologies de e-commerce (Mails, SSL, data mining, J2ME, Web) : projet "Inpres-Ferries"- suite (2)

3<sup>ème</sup> Informatique de gestion  
2011-2012



Claude Vilvens et Ludovic Kuty  
- en collaboration avec Christophe Charlet



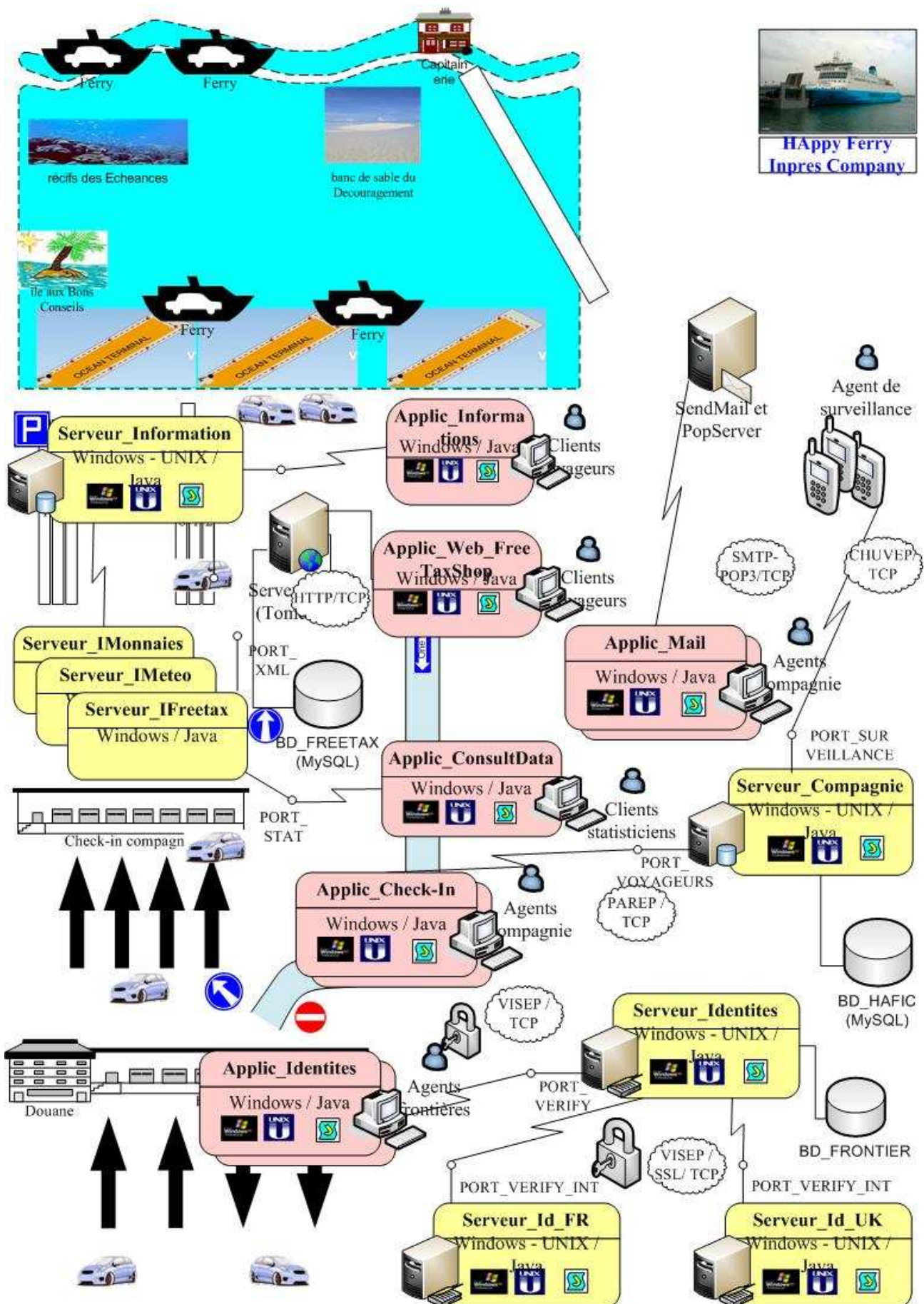
### 1. Le contexte

Il est le même que celui du laboratoire de "Réseaux et technologies Internet", à savoir celui de la société "**Happy Ferry Inpres Company**" (**HAFIC**) qui exploite des lignes de ferry entre la France, la Grande-Bretagne et la Scandinavie.

### 2. Règles d'évaluation du laboratoire

Il convient de référer à l'énoncé de la première partie de ce laboratoire. Pour rappel :

- ♦ La deuxième partie de ces travaux (statistiques avec JFreechart, mails, SSL) sera évaluée par l'un des professeurs du laboratoire à partir du 5 décembre 2011 (avec rentrée d'un dossier papier limité - le délai est toujours à *respecter* impérativement).
- ♦ La troisième partie (J2ME, WEB, Java Web Start, Struts, ...) sera évaluée lors de l'examen de laboratoire en janvier-février 2012 (le dossier papier n'est plus nécessaire).



## Les travaux de l'évaluation 2

### 1. Les informations statistiques

Le serveur **Serveur\_IFreetax**, évoqué dans la première partie, va se voir doté de fonctionnalités complémentaires importantes. En fait, il a accès à la base de données BD\_FREETAX qui contient, outre les catalogues de produits, les chiffres de ventes sur les ferries (qu'il s'agisse de ventes directes ou de ventes par réservation – voir application Web Applic\_Web\_FreeTaxShop ci –dessous).

Plus précisément, la base contient, au minium et sans entrer dans les détails, les tables suivantes :

Producteurs	<i>commentaires</i>
NumProducteur	PK
Nom	
AdressePostale	
AdresseEEmail	
DateDernièreCommande	
NumeroCompteIBAN	

Produits	<i>commentaires</i>
NumProduit	PK - ex: AlcFR5021-75
Catégorie	Alcool, Parfumerie, Tabac
NomProduit	ex: Cognac Canabis, Cigares Teufteuf
Pays	code ISO : FR, UK, BE
NumProducteur	FK
Conditionnement	Bouteille, Flacon, Boîte, Pochette
QuantitéUnitaire	ex: 75 cl, 40 g
QuantitéPackage	ex: 1 (cognac), 20 (cigares), 6 (bières)
PrixUnitairePackage	ex: prix d'une bouteille (cognac, prix d'un pack de 6 bières, ...
UnitéMonétaire	code ISO : EUR, USD, GBP,

Stocks	<i>commentaires</i>
NumProduit	FK
QuantitéEnStock	exprimée en termes de packages
Seuil	seuil de rupture de stock
DateDernièreCommande	
QuantitéEnCommande	
NumProducteur	FK

Ventes	<i>commentaires</i>
NumProduit	FK
Quantité	exprimée en termes de packages
DateVente	
UnitéMonétairePaiement	code ISO : EUR, USD, GBP,
NumRéservation	NULL si achat direct non réservé
NationalitéAcheteur	code ISO : FR, UK, BE
AnneeNaissanceAcheteur	ex: 1982, 1960

ainsi qu'une table **Users** permettant le contrôle des utilisateurs.

Le serveur **Serveur\_IFreetax** est consulté sur le port **PORT\_STAT** au moyen d'une application **Applic\_ConconsultData**, dont le rôle est de demander le traitement des informations disponibles dans **BD\_FREETAX**. On y utilise le protocole **IDP** (**I**nformation and **D**ecision **P**rotocol) dont les commandes sont :

protocole <b>IDP</b>		
<b>Commande</b>	<b>Sémantique</b>	<b>réponse éventuelle</b>
LOGIN	démarrage de l'application : un comptable se fait reconnaître <i>paramètres</i> : nom, password	oui ou non – validation au moyen d'un digest dans la base <b>BD_FREETAX</b>
GET_STATISTICS_DESCR	demande du nombre d'articles d'une catégorie vendus par jour pour une semaine ou un mois donné, avec calcul de la moyenne, du mode et de l'écart-type de la série statistique "nombre d'articles d'un type donné vendus par jour") <i>paramètres</i> : la catégorie, la semaine ou le mois –les données du mois précédent si paramètre absent	les données brutes et les paramètres statistiques demandés
GET_GRAPH_1D	demande d'un graphique statistique réalisé à l'aide de la librairie <b>JFreechart</b> : diagramme sectoriel ou histogramme pour un article selon les jours (la commande <b>GET_STATISTICS_DESCR</b> doit donc d'abord être envoyée) <i>paramètres</i> : la catégorie, la semaine ou le mois + type de graphique = SEC, HIST	, histogramme comparé pour des articles de même catégorie mais de marque différente, série chronologique pour un article sur une semaine, corrélation vente d'un article/température moyenne mensuelle
GET_GRAPH_1D_COMP	demande d'un graphique statistique réalisé à l'aide de la librairie <b>JFreechart</b> : histogramme comparé pour des articles de même catégorie mais de marque différente <i>paramètres</i> : la catégorie, la semaine ou le mois + type de graphique = HIST_COMP	
GET_GRAPH_1D_CHRON	demande d'un graphique statistique réalisé à l'aide de la librairie <b>JFreechart</b> : série	

	chronologique pour un article sur une semaine <i>paramètres</i> : la catégorie, la semaine ou le mois + type de graphique = CHRONO	
GET_GRAPH_2D_CORR	demande d'un graphique statistique réalisé à l'aide de la librairie <b>JFreechart</b> : corrélation vente d'un article/âge de l'acheteur <i>paramètres</i> : la catégorie, la semaine ou le mois + type de graphique = CHRONO	
TEST_COMP	demande un test d'hypothèse de comparaison des moyennes du nombre de ventes par semaine d'un article de catégorie et de marque données selon que deux nationalités de l'acheteur <i>paramètres</i> : l'identification de l'article, le seuil utilisé	différence significative ou non

## **2. La messagerie électronique**

Au niveau des frontières, on aura pu constater que le dialogue impliqué par le protocole **VISEP** entre le serveur **Serveur\_Identités** et son application cliente **Applic\_Identités** est à sens unique : c'est essentiellement le client qui envoie des requêtes, sauf dans le cas particulier du TIME\_OUT.

Cependant, il peut arriver que les autorités compétentes (frontier.gov) souhaitent envoyer des messages aux agents des frontières, messages soit anodins (du genre "Prévoyez d'ores et déjà le rôle des soirées de réveillon") ou plus importants ("La personne suivante est recherchée et ne peut passer la frontière : xxx, yyy, ...).

Tout ceci n'a rien à voir avec VISEP et ce sont les e-mails qui seront utilisés. Dans ce contexte :

**a)** D'une part, il s'agit tout d'abord de développer une petite application **Applic\_Mail** permettant de traiter le courrier électronique relatif aux informations échangées entre frontier.gov et les agents des frontières. Plus précisément, elle comporte une fonctionnalité GUI **Java** de gestion classique d'e-mail (type "mini-outlook"), permettant d'envoyer et de recevoir un mail

- soit simple de type texte;
- soit composite avec des pièces attachées qui sont des images gif ou jpg, ou encore un paquet de bytes (tiens ?) ou même des objets sérialisés (instance de la petite classe **PieceAttachee** qui se limite à encapsuler les composantes de l'information, selon sa nature).

L'utilisateur devrait idéalement être prévenu dans un délai de 5 minutes de l'arrivée d'un nouveau message. A cet effet, l'application utilisera un thread de polling qui "interrogera" périodiquement la boîte aux lettres.

b) La nature de message est fixée dans le sujet du message. Epinglons (la liste peut être complétée selon les besoins du dialogue réel) :

- ◆ FRONTIER\_INFO : simple texte, accompagné ou non de pièces attachées (genre image d'un carton d'invitation);
- ◆ FRONTIER\_INFO\_ACK : clair
- ◆ FRONTIER\_WANTED : notification qu'une personne d'identité donnée ne peut quitter le pays; bien sûr, on tentera d'implémenter le maximum de sécurité (chiffrement de l'identité, authentification de l'émetteur de la recherche;
- ◆ FRONTIER\_WANTED\_ACK : clair
- ◆ FRONTIER\_WANTED\_TOO\_LATE : clair aussi : la personne recherchée est déjà passée. Dans ce cas, il reste aux agents des frontières transférer un ultime appel aux agents portuaires : peuvent-ils arrêter un véhicule de plaque minéralogique donnée ? Ceci va se faire avec l'application suivante :

### 3. Les communications sécurisées par SSL

On se souviendra que l'application **Applic\_Identités** assure de manière sécurisée la vérification des identités des voyageurs (protocole VISEP). Mais la commande **VERIFY\_ID**, adressée au serveur **Serveur\_Identités** est supposée porter sur une personne de la nationalité du port (donc, par exemple, des Belges si le port est en Flandre). Mais pour le cas où les voyageurs sont de nationalité différente, il faut utiliser plutôt une nouvelle commande de VISEP : **VERIFY-ID\_INTERNATIONAL**, qui est sécurisée au moyen de TLS/SSL, et qui s'adresse à d'autres serveurs nationaux du même type que **Serveur\_Identités** (**Serveur\_Id\_FR**, **Serveur\_Id\_UK**, ...). En fait, ce n'est pas l'application cliente qui s'adresse à ces serveurs, mais le **Serveur\_Identités** qui fait office de "passerelle".

protocole <b>VISEP</b> (suite)		
commande	sémantique de la requête	réponse éventuelle
VERIFY-ID_INTERNATIONAL	pour vérifier l'identité d'un voyageur <i>paramètre</i> : nom, prénom et numéro de carte d'identité du voyageur, nationalité (format ISO standard des pays : BE, FR, UK, US, etc), le tout crypté avec la clé de session + vérification de la signature de la réponse	VERIF_INT_SUCCESSFULL ou VERIF_INT_FAILED, réponse signée par le serveur

La seule différence avec **VERIFY\_ID** est, outre la spécification de la nationalité, le fait que, du côté serveur, il y mise en route (ou récupération) d'une session SSL : SSL-handshake puis SSL-record.

Les certificats nécessaires seront fabriqués avec OpenSSL et/ou la programmation. Les keystores seront visualisés avec Keytool IUI. On prendra en charge au minimum deux serveurs internationaux distincts.

**En pratique** : Pour les points 2 et 3, il convient d'analyser les trames au moyen d'un sniffer quelconque.

### Les travaux de l'évaluation 3

#### 4. J2ME

Les installations portuaires couvrent une surface de taille assez importante. A priori, les seuls véhicules qui s'y déplacent, mis à part ceux du service, sont ceux des voyageurs qui ont passé avec succès le poste frontière et le check-in de la compagnie. Cependant, il peut arriver, par exemple, que :

- ♦ des véhicules de voyageurs égarés ne se trouvent pas du tout où ils devraient être (pas dans les files attribuées à une compagnie pour un ferry donné, tranquillement parké sur le trottoir de la salle d'attente, etc);
- ♦ des véhicules suspects aient pénétré la zone (profitant de failles dans la clôture ou escortant soi-disant un véhicule de livraison) et tentent de monter discrètement sur un ferry à la faveur de la confusion des opérations d'embarquement.

Les autorités du port ont donc mis en places des agents de surveillance dont le rôle est de parcourir constamment les installations, particulièrement les zones peu fréquentées. Lorsqu'ils détectent la présence inexplicquée d'un véhicule, ils utilisent sur le smartphone dont ils sont équipés une application J2ME **Applic\_Control\_Vehicles** qui va converser avec **Serveur\_Compagnie** qui attend sur un port dédié PORT\_SURVEILLANCE et qui utilise le petit protocole **CHUVEP** (CHeck Unexpected VEHicle Protocol)

protocole <b>CHUVEP</b>		
Commande	Sémantique	réponse éventuelle
LOGIN	démarrage de l'application : un agent de surveillance ouvre une session en utilisant son numéro de matricule <i>paramètres</i> : numéro de carte	oui ou non – validation dans la base BD_HAFIC
CHECK_VEHICLE	demande de vérification du passage au check-in d'un véhicule <i>paramètres</i> : numéro de plaque et nationalité du véhicule	♦ si le véhicule est retrouvé : nom du ferry et numéro de ligne correspondant ♦ si non : NOT_FOUND
<b>a) pour un véhicule en ordre :</b>		
VEHICLE_RIGHT_PLACE	après avoir ramené le véhicule à sa place, l'agent de surveillance le signale <i>paramètres</i> : numéro de plaque du véhicule	ACK applicatif et déconnexion
<b>b) pour un véhicule pas en ordre :</b>		
RECORD_VEHICLE	les coordonnées du véhicule sont à la fois enregistrées sur le mobile et envoyé à Serveur_Compagnie - l'agent de surveillance n'est pas habilité à contrôler des identités : il se contente (si	les coordonnées sont mémorisées dans BD_HAFIC et un mail urgent est envoyé au poste de police situé à l'entrée de la zone portuaire



	possible) de bloquer le véhicule jusqu'à l'arrivée des forces de police <i>paramètres</i> : numéro de plaque, marque, modèle, nationalité du véhicule	
POLICE_CHECK_BEGIN	envoyé dès que les forces de police prennent la relève	ACK applicatif et déconnexion
En fin de journée :		
LIST_RECORDED_VEHICLES	envoi de la liste complète des véhicules suspects au Serveur_Identités	ACK applicatif et déconnexion

## **5. Applic Web FreeTaxShop**

Il s'agit ici de faciliter l'achat de produits freetax lors de la traversée en permettant aux voyageurs qui ont déjà réservé leur traversée de réserver d'autres et déjà de tels produits. Lors de la traversée, il leur suffira, une fois le ferry entré dans les eaux internationales, de passer retirer leur commande au shop du ferry. Cette application **Applic\_Web\_FreeTaxShop** est bien sûr une application Web MVC.

### **5.1 L'internationalisation**

L'internationalisation va être prise en compte afin de permettre aux clients étrangers de fréquenter sans difficulté le shop du ferry. On imagine donc que l'application WEB démarre par une JSP dédiée au **login et choix de langues**. Le client va contacter le site WEB en invoquant le Java Server Page **Contexte.jsp** : il reçoit ainsi une page WEB de type formulaire – en fait, c'est une **applet signée**. Ce formulaire comporte une boîte de listes contenant

- ◆ la liste des langues utilisables sur le site;
- ◆ des zones d'entrée pour son nom, son code de réservation et son adresse e-mail;
- ◆ un bouton "Connect".

Les langues utilisables se trouvent dans une table *Langues* d'une base BD\_FREETAX, également dépositaire des stocks de produits à vendre. L'internationalisation sera réalisée pour les JSPs suivants, et pour les messages courants, en utilisant la technique des "bundles".

Toutes les pages suivantes utiliseront la balise personnalisée sans corps **<status:date-heure>** pour réaliser dans chaque JSP l'affichage (dans la langue choisie) de la date et heure.

### **5.2 Les achats**

On s'en doute, les clients effectuent des réservations selon la technique du "**caddie virtuel**" dans une interface graphique qui illustre très schématiquement les produits disponibles. La démarche consiste à se promener dans les pages catalogues du site et à choisir au fur et à mesure des éléments. Tout comme pour le laboratoire de "Réseaux et technologies Internet", on attend une gestion cohérente des stocks (ne pas vendre deux fois le même produit, ne pas admettre qu'un même produit puisse se trouver dans deux caddies différents, etc).

### **5.3 Le paiement des achats**



La page de confirmation-paiement des achats est une JSP dédiée à la **confirmation des achats et introduction des données complémentaires**. Cette JSP utilise la balise avec corps `<status:list>` qui affiche dans le JSP la liste des achats effectués. Mais surtout, le paiement, après introduction du numéro de carte de crédit, va provoquer la mise en route d'une séquence sécurisée (comment ?) : les informations de paiements (nom client, numéro de carte, montant) sont envoyées à un SERVEUR\_BANQUE sur son port PORT\_PAY et, bien sûr, le serveur va vérifier si le paiement peut être effectué ou pas au moyen d'une base de données BD\_COMPTEs. Le protocole qu'il convient ainsi de définir est désigné par **BDP** (Bank Dialog Protocol). En cas de succès, une facture est envoyée par mail au client. Cette facture va servir de preuve au client pour aller rechercher ses achats au shop du ferry.

#### **5.4 La réponse et la clôture de la transaction (réussie ou non)**

Quelle que soit l'issue de la transaction, le client reçoit une dernière JSP visant la **clôture des opérations**. Outre le résultat, cette page affiche le temps d'utilisation total de l'application au moyen de la balise sans corps `<status:using-time>`.

### **6. Java Web Start**

On demande que l'application fenêtrée développée dans la première partie du laboratoire (XML – elle permet au voyageur qui l'utilise de demander plusieurs types d'informations) devienne une application Java Web Start.

Est-ce également envisageable pour l'application Applic\_Mail du point 2 ci-dessus ?

### **7. L'application Applic Web FreeTaxShop – release 2.0**

Le site évoqué au point 5 devrait à présent évoluer, quant à son application Web, en une version 2.0 utilisant les mécanismes de la plate-forme Struts. Quels changements faut-il apporter ? Peut-on les qualifier de fondamentaux ou de légers ?

Soyez créatifs et imaginatifs ... mais restez rationnels et raisonnables, notamment quant aux informations utilisées ...

s: CV & LK – avec le concours de CC.

