

## Module de Système d'Information – Chapitre Informatisation du Système d'Information

## TD : Intergiciel et topologie des Systèmes d'Information hétérogènes

L'arrière boutique (*back-office*) d'une entreprise peut se gérer avec un seul Progiciel de Gestion Intégrée (PGI, ou ERP en anglais) qui intègre la gestion de toutes les activités de l'entreprise dans un seul logiciel. Dans le cas contraire, le Système d'Information de l'entreprise sera une intégration de plusieurs progiciels de gestion identifiés comme étant chacun identifié comme le meilleur de sa catégorie (*best-of-bread*) pour gérer un besoin particulier de l'entreprise. Ce TD se concentre sur le second cas.

**Exercice 1 : Les activités de l'entreprise**

Question 1 : Rappelez les activités classiques de l'entreprise et les besoins de gestion associés.

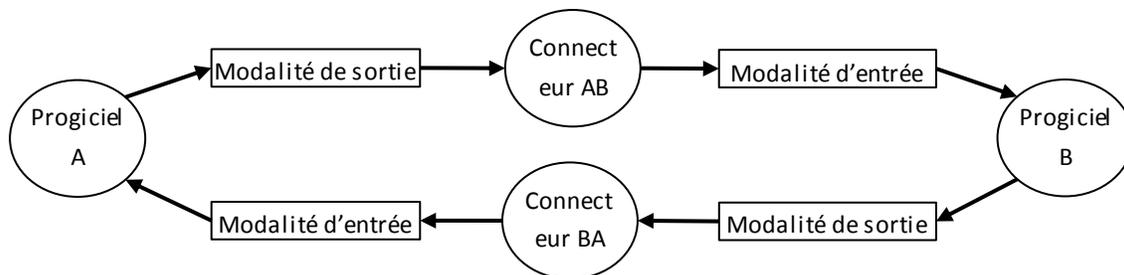
Question 2 : Pour chacun des progiciels de la liste ci-après, associez le ou les besoin(s) de gestion qu'il permet de gérer parmi ceux évoqués précédemment.

Liste (non-exhaustive) de logiciels de gestion pour l'entreprise :

- |                |               |                         |                            |
|----------------|---------------|-------------------------|----------------------------|
| 1. Ciel        | 10. Eureka    | 19. Cogivea             | 28. Adequasys Allegro      |
| 2. Concurive   | 11. Windchill | 20. Acteos              | 29. Creasoft Stocknet      |
| 3. Clipper     | 12. Sage 100  | 21. Evoliz              | 30. Yourcegid Retail       |
| 4. Lascom      | 13. Audros    | 22. vTiger              | 31. ABW C-FIRST            |
| 5. Prestashop  | 14. Fletesia  | 23. Magento             | 32. Apisoft Optimaint      |
| 6. SugarCRM    | 15. GLPI      | 24. GCS TMS             | 33. Qubes XXL              |
| 7. EBP         | 16. Sellsy    | 25. Divalto Idylis      | 34. Elfa-systèmes Clarisse |
| 8. Solune Alta | 17. Pentaho   | 26. Siveco Group Coswin | 35. SolidWorks PDM         |
| 9. Knowage     | 18. Akanea    | 27. Hardis Reflex       | 36. Salesforce.com         |

**Exercice 2 : Connecter deux logiciels**

Le but est de permettre l'échange bidirectionnel de données entre deux progiciels qui ne sont pas prévus pour fonctionner ensemble. Pour ce faire, des programmes de conversion de données (fichiers, base de données, flux, ...) supplémentaires sont créés pour rendre ces échanges possibles. Ces programmes sont appelés des connecteurs. Ces petits programmes sont souvent des scripts BATCH, SHELL ou encore PHP, qui vont commencer par lire/extraire/recevoir (selon la modalité) une donnée provenant du logiciel source, puis l'analyser, la traiter, la convertir, et enfin écrire/charger/envoyer (selon la modalité) le résultat de la conversion à destination de l'autre logiciel.



Les six modalités d'entrée et de sortie les plus courantes sont :

<i>flux entrant</i>	<i>flux sortant</i>
1. Lire dans un fichier (csv, xml, ...)	4. Écrire dans un fichier (csv, xml, ...)
2. Extraire depuis une base de données	5. Charger dans une base de données
3. Envoyer un message via webservice (XML-RPC, REST,...)	6. Recevoir un message via webservice (XML-RPC, REST,...)

Le déclenchement d'un connecteur peut se faire :

- Ponctuellement et manuellement par l'utilisateur
- Ponctuellement et automatiquement par un programme (qui, par exemple, surveille l'apparition d'un évènement, procédure stockée (*trigger*), ...)
- Périodiquement et automatiquement, toutes les  $X$  secondes, chaque heure, chaque nuit, ... par le système d'exploitation (et le service de planification de tâches CRON, par exemple) ou par programmation interne dans le connecteur (qui boucle à l'infini).

Question 1 : En considérant uniquement les six modalités ci-avant (trois en entrée, trois en sortie), représentez schématiquement les 9 cas possibles de connecteur AB entre deux logiciels. Indiquez à chaque fois par un verbe le travail effectué par le connecteur sur la ressource intermédiaire entre lui et le logiciel.

Question 2 : En considérant uniquement les trois types de ressources ci-avant (fichiers, base de données et webservice), combien de configurations sont possibles pour connecter deux applications A et B ?

### Exercice 3 : Connexion de plusieurs applications en mode point-à-point

Nous allons étudier le nombre de connecteurs nécessaires pour que plusieurs applications puissent toutes communiquer entre elles. Pour cela, nous appelons la théorie des graphes pour modéliser le problème :

La topologie logicielle est représentée par un graphe orienté, où les sommets du graphe sont les logiciels de l'entreprise et les arêtes du graphe sont les connecteurs (permettant l'échange unidirectionnel d'information d'une application vers une autre).

Nous considérons que le graphe de la topologie logicielle est un graphe complet (ou encore « *plat de spaghettis* »).

Rappel : un graphe complet est un graphe dont tous les sommets sont reliés entre eux.

Question 1 : Dessinez les graphes complets modélisant les cas de 2, 3 et 4 logiciels.

Question 2 : Combien d'arêtes comptent chacun de ces graphes ?

Question 3 : Redessinez le cas de 4 logiciels et ajoutez-y un cinquième logiciel. Combien de nouvelles arêtes doivent être ajoutées pour que le graphe reste un graphe complet ?

Question 4 : Déduisez (de la question précédente) la règle de calcul du nombre de nouveaux connecteurs à créer/installer/maintenir à chaque ajout ou changement de logiciel ?

Question 5 : Pour un Système d'Information de N logiciels, combien de connecteurs doivent être créés/installés/maintenus pour que chacun de ces logiciels puissent communiquer avec tous les autres ?

#### **Exercice 4 : Enterprise Application Integration (EAI)**

Dans une approche EAI (qui fait partie intégrante des démarches d'urbanisation des Systèmes d'Information), un logiciel supplémentaire sert de concentrateur (hub). Chaque logiciel communique alors avec les autres en passant obligatoirement et uniquement par le logiciel concentrateur. Ainsi, chaque logiciel doit être connecté au logiciel concentrateur. Nous parlons alors de topologie logicielle « *en étoile* ».

Question 1 : Dessinez les graphes modélisant les cas de 2, 3 et 4 logiciels.

Question 2 : Combien d'arêtes comptent chacun de ces graphes ?

Question 3 : Redessinez le cas de 4 logiciels et ajoutez-y un cinquième logiciel. Combien de nouvelles arêtes doivent être ajoutées ?

Question 4 : Déduisez (de la question précédente) la règle de calcul du nombre de nouveaux connecteurs à créer/installer/maintenir à chaque ajout ou changement de logiciel ?

Question 5 : Pour un Système d'Information de N logiciels, combien de connecteurs doivent être créés/installés/maintenus pour que chacun de ces logiciels puissent communiquer avec tous les autres ?

Question 6 : Une topologie logicielle EAI est un projet complexe à mettre en œuvre et demande un gros investissement de départ. Pensez-vous que ce type de topologie logicielle puisse être mise en place dans de petits systèmes d'information avec peu de logiciels ?

## Exercice 5 : Cas d'application

*L'objectif pédagogique de cet exercice est d'appliquer les notions vues depuis le début du cours en prenant le prétexte de produire des documents de conception pour un système d'information, en identifiant les flux d'information et en distinguant les progiciels et l'intergiciel.*

*Consignes : Pour ce travail, associez-vous en binômes (au besoin, une seule équipe de 3 sera acceptée). Chaque binôme déposera son travail sur Moodle à la fin de la séance.*

1) **LE BESOIN.** Imaginons une entreprise de sous-traitance d'équipement automobile en B2B dont l'activité inclue de la production et du bureau d'étude (ex : Caromar Techno, Valeo, Delphi, Bosch ...). Les clients sont aussi bien des fabricants automobiles (Peugeot, Renault, ...), des garagistes indépendants (AD, ...), que des magasins généralistes (Feu Vert, Norauto, ...). Les produits conçus et fabriqués sont aussi bien des pièces d'origine que des pièces adaptables.

Listez dans un premier tableau les activités des départements de cette entreprise, qui comprennent au moins : un service commercial, une chaîne de production, une chaîne d'approvisionnement, des entrepôts, un service de livraison, un bureau d'étude et un contrôle qualité (vous êtes libre d'en ajouter d'autres). Listez dans un deuxième tableau les différents métiers et les effectifs de chaque service. Listez dans un troisième tableau les locaux (bâtiments, bureaux, succursales), les clients et les partenaires de cette entreprise (fournisseurs, donneurs d'ordres).

Activités	Départements	Métiers	Effectifs	Locaux	Clients	Partenaires
...	...	...	...	...	...	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...

La suite de l'exercice est maintenant de répondre aux besoins de gestion de cette entreprise par la mise en place d'un système d'information.

2) **LES FLUX D'INFORMATION.** Pour chaque activité de l'entreprise, représentez graphiquement les entités de l'entreprise (acteurs, progiciels, dispositifs matériels) et les informations que vous pensez qu'elles peuvent produire ou dont elles peuvent avoir besoin. Ces entités peuvent être source d'une information ou destinataire d'une information : faites apparaître les flux d'information et les ressources permettant de les gérer entre ces entités (cf. schémas utilisés lors de la restitution des lectures de témoignages).

3) **LES PROGICIELS.** Choisissez parmi les progiciels de gestion (approche « Best of breads ») proposés dans le catalogue<sup>1</sup> d'un prestataire pour chaque activité et besoin de gestion. Vérifiez si la couverture fonctionnelle répond au besoin et listez les fonctionnalités ou données éventuellement manquantes (afin de requérir des précisions auprès du prestataire ou d'identifier les développements supplémentaires à estimer).

<sup>1</sup> URL du catalogue du prestataire : <http://www.guillaumeriviere.name/estia/si/catalogue/>

4) **L'INTERGICIEL (EAI)**. En fonction des interfaces d'échange de chaque progiciel, cartographiez les connecteurs ou les concentrateurs à développer et à paramétrer.

(1) Source			(2) Traitement		(3) Destination		
Progiciel	Format en sortie	Données en sortie	Calcul, conversion, message, trigger ...	Langage de programmation utilisé	Progiciel	Format en entrée	Données en entrée
...	...	...	...	...	...		...
...	...	...	...	...	...		...
...	...	...	...	...	...		...

5) **SYNTHÈSE**. Produisez un nouveau schéma, de votre système d'information, représentant les acteurs, les logiciels, les connecteurs, les concentrateurs, les échanges d'information et les bases de données. Listez dans un tableau les ressources logicielles, et estimez un budget de base incluant les coûts d'achat de licences et de règlement d'abonnements.