

INDUSTRIALISATION ET RATIONALISATION

A. LA PROBLEMATIQUE

La mission de toute production informatique est de délivrer le service attendu par les utilisateurs. Ce service se compose de résultats de traitements et de la mise à disposition de ressources informatiques permettant de faire fonctionner des applications. Pour ce faire, il incombe à la production de maîtriser en permanence la qualité du service offert en sachant s'adapter aux nouveaux besoins des utilisateurs ou de fournir un service existant à de nouveaux utilisateurs.

Les besoins de toutes les productions sont universels, qu'il s'agisse des fonctions usuelles d'arrêts et démarrages des services, de configuration des services, de sauvegardes et restaurations des services, de purges des fichiers de suivis, de mesure de charge des ressources, de mesure de la qualité du service

Cette liste non exhaustive montre simplement que le quoi faire est connu, il s'avère dans certains cas ne pas être formalisé.

En outre, les progiciels utilisés en général ne décrivent pas les processus métier : ce sont avant tout des outils mis à disposition de nos professionnels pour faciliter l'exercice de leur métier mais non pour formaliser les processus métier.

Les données utilisées par ces progiciels sont rarement communes, à part peut-être les données relatives aux équipements pour les solutions à base de « Framework ».

On ne peut que constater que cette situation se rapproche de la situation des années 80 pour ce qui concernait les autres métiers de l'entreprise :

- Pas de modélisation des environnements de production
- Des données dynamiques (Process, jobs...)
- Des données de nature hétérogène (Fichier, process,..)

- Pas de véritable « système d'information » à l'usage des producteurs

Pour atteindre cet objectif de mise en qualité, plusieurs voies sont possibles, et la démarche BV Associates préconise d'agir sur les éléments constitutifs du niveau de service : la qualité technique, la maintenabilité, le savoir-faire métier en utilisant la démarche méthodologique IMEPSIA™ et les outils qui l'accompagnent afin d'industrialiser les environnements de production.

Cette industrialisation repose sur un principe de mutualisation des procédures techniques, c'est à dire dans le développement et la mise en place de composants industriels génériques et réutilisables.

Cette mutualisation permet de diffuser et de partager entre les individus constituant les équipes de production, des connaissances techniques et un savoir-faire commun. Par ailleurs l'utilisation de composants et procédures communes offre, le cas échéant, la possibilité de mutualiser les ressources et compétences humaines.

En outre, ce principe de mutualisation technique et humaine a pour conséquence d'impacter les coûts d'intégration et d'exploitation des environnements et d'une manière générale sur le TCO en permettant de maîtriser :

- Coûts et délai de mise en oeuvre des évolutions de version
- Coûts de maintenance.
- Coûts de personnel en intégration (mutualisation du savoir-faire)
- Coûts de personnel de suivi de production (une même personne est en mesure d'assurer le suivi d'un plus grand nombre de clients en simultané).
- Coût de formation : une fois formé à la méthode, la formation consiste principalement à prendre la documentation des composants pour avoir une bonne idée du fonctionnement du composant logiciel d'exploitation.

B. AU CŒUR DE L'INDUSTRIALISATION : LES I-TOOLS (NOYAU D'I-SIS)

Caractéristiques

L'utilisation d'outils d'intégration permet de normaliser les développements des analystes d'exploitation, des ingénieurs système et de tous les experts ayant en charge un composant de l'architecture d'exploitation. Les I-TOOLS proposés par BV Associates permettent de décrire les objets du système de façon unique et de voir les machines comme un véritable système d'information homogène pour l'ensemble des équipes DSI. Ils constituent une véritable plateforme de développement de scripts normalisés. Ils permettent également d'isoler naturellement les données de configuration dans des tables parfaitement identifiées.

La description par les I-TOOLS d'un service applicatif, et son intégration constituera un I-CLES, composant logiciel d'exploitation des services. Ce composant hérite de toute la puissance de description et de développement des I-TOOLS ce qui lui assure une structure homogène, normalisée, évolutive, portable et réutilisable.

Pour une application ou un logiciel, développer un I-CLES, c'est entre autre, décrire :

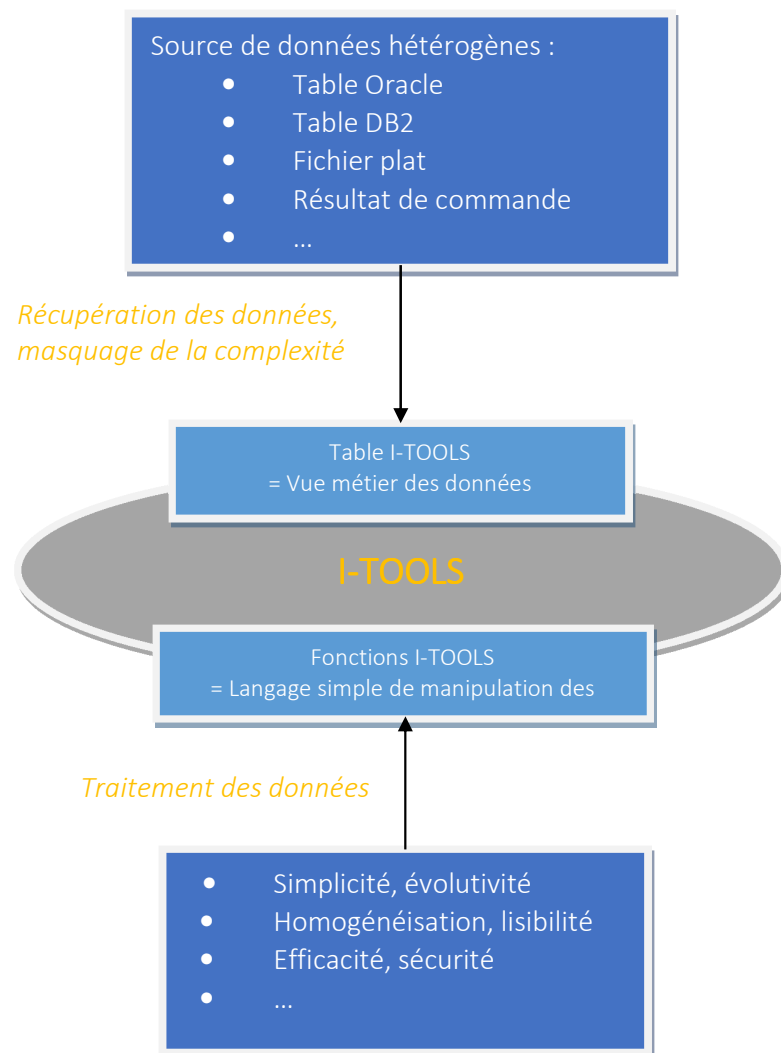
- Les tables des objets et leurs relations (intégrités, cohérences, parcours, ...)
- Les tables de configuration d'un service applicatif
- Les fonctions de contrôle standard (statuer, démarrer, arrêter, importer, extraire ...)
- Les fonctions d'exploitation standards (logs, purges, sauvegardes, ...)
- Les représentations graphiques des objets

Les I-TOOLS mettent à disposition une structure *Table*, permettant d'homogénéiser et de décrire de manière évoluée, le format des données du système d'information à manipuler. Ils nous dégagent ainsi de toute la complexité liée à la source de données (serveur, base, fichier, résultat, ...), son format, son support, ...

Une table correspond à une vue du système d'information, que l'on peut ensuite interroger et mettre à jour, via des fonctions standard I-TOOLS dont les caractéristiques sont les suivantes :

- Standardisation de l'accès aux systèmes d'informations, quelques soient les machines et leur OS - Unix Linux et Windows
- Standardisation de l'accès au système d'information de l'environnement d'exploitation, quelque soit la source des données et son format (fichier, table Oracle, résultat de commande, ...)
- Fonctions de manipulation de tables de type SQL très intuitif
- Simplification des traitements
- Robustesse et efficacité

Le schéma ci-dessous représente l'architecture logique des I-TOOLS.



D'autre part, les I-TOOLS s'agrémentent de fonctionnalités nécessaires à toute exploitation informatique :

- Des fonctions de logs et de traces standard.

- Des fonctions de sécurisation des données.
- Des fonctions d'exécution sécurisée d'un traitement sous un utilisateur donné.

L'apport d'I-SIS : Console de monitoring et de pilotage sécurisé

- Normalisation, centralisation et consolidation des logs générés dans un format matriciel avec fonctions d'accès en SQL-like
- Maîtrise et analyse facilitées du système d'information d'exploitation : fichiers de configuration avec accès aux informations systèmes en mode SQL-like
- Outil de livraison technique, simple et éprouvé sur des environnements de production, paramétrable dans des fichiers de configuration
- Visualisation homogène et affranchissement des contraintes spécifiques à chaque plate-forme hétérogène Unix, Linux et Windows
- Capitalisation dans le référentiel des I-CLES – cf. plus haut
- Outillage technique commun et fédérateur
- Respect assuré des règles de normalisation

La console I-SIS

Pour les besoins du pilotage de la production, la console WEB offre :

- La visualisation et le suivi des traces de production - applicatives ou techniques
- Le suivi et la manipulation des livraisons - journaux
- Une IHM sécurisée : définition et gestion fine des droits
- Des fonctions d'édition - copier, couper, coller..
- Des fonctions d'exploration des services et de leurs composants en exploitation : explorer, ouvrir, fermer...

En outre la console fournit :

- des commandes et des scripts standard, comme l'arrêt/démarrage d'un service et le calcul de son état : arrêté, démarré...
- l'encapsulation des procédures complexes pour le personnel moins expert
- la prise en compte dynamique de la mise à jour d'un service
- la vue globale et détaillée du système d'information et des applications

C. GAINS OBTENUS

Le fait d'organiser les scripts en composants logiciels d'exploitation réutilisables dans tous les environnements permet de transformer le savoir-faire individuel en savoir-faire d'entreprise et rend ainsi le personnel plus polyvalent.

La constitution de composants logiciels d'exploitation normalisés – les I-CLES - permet à des techniciens n'ayant pas travaillé sur un composant technique son appropriation très rapide. Chacun peut comprendre le composant à la seule lecture de ses tables de configuration et de modélisation. Le savoir-faire codé dans le composant logiciel d'exploitation devient alors accessible à tous et constitue de ce fait un véritable savoir-faire d'entreprise.

Par ailleurs, la constitution de composants logiciels d'exploitation permet de capitaliser le savoir-faire d'entreprise dans un composant technique unique connu de tous. La réutilisation systématique du composant dans des contextes variés garantit la fiabilité du composant. En cas de détection d'un problème sur le composant, la solution apportée peut immédiatement bénéficier aux autres environnements.

En outre le chiffrage du coût d'intégration est facilité par l'identification des composants déjà intégrés et de ceux à intégrer.

L'approche composant permet également de mesurer rapidement l'impact d'une nouvelle version en identifiant la liste des composants touchés.

De plus, si ces composants logiciels d'exploitation sont réalisés en utilisant des outils techniques identiques pour tous, la configuration et la maintenance de ces composants en est grandement facilitée, d'où une plus grande productivité.

Les gains obtenus sont donc à la fois techniques :

- Réduction des délais de mise en production,
- Réduction des charges d'intégration,
- Plus grande disponibilité des « services » rendus aux utilisateurs,
- Meilleure qualité du suivi,
- Réduction du temps de traitement des incidents,

- Réduction des charges de mise en œuvre et d'exploitation,
- Meilleurs échanges entre les différents acteurs,

et financiers, à savoir :

- Diminution des coûts de mise en oeuvre des évolutions de version
- Diminution des coûts de maintenance,
- Diminution des coûts de personnel d'intégration et de suivi de production par la mutualisation,