

Informix.
software

Informix Newsletter

2nd trimestre 2012

User Group Informix France

Bienvenue dans la Newsletter du User Group Informix France

Au programme

Editorial	3
Evénement : Retour sur l'IIUG à San Diego, Californie.....	4
Technical Tip: Nouvelles fonctionnalités en 11.70.xC5	5
Technical Tip: Nouvelle fonctionnalité IFX_BATCHEDREAD_INDEX	7
Technical Tip: Enterprise Replication – Pourquoi l'utiliser ?	8
Technical Tip: Les applications Informix-4GL ont-elles un avenir ?	9
Derniers articles	14
Vidéos	14
Liens utiles.....	14
Informix blogs	14
Abonnement / Annulation / Avis	15
Les contributeurs de ce numéro	15

Editorial



La planète Informix continue à nous faire rêver. L'innovation est le point clé du moteur Informix.

Les utilisateurs et fans d'Informix qui ont pu assister à la conférence annuelle de l'IIUG à San Diego du 22 au 25 avril 2012 ont pu découvrir les nouveautés et l'enthousiasme autour de l'offre Informix d'IBM.

Informix IDS (Informix Database Server) continue son évolution avec la sortie de la version 11.70.xC5 au mois de mai 2012 qui a apporté son lot de nouveautés.

IWA – Informix Warehouse Accelerator a un an et une nouvelle version vient de voir le jour avec de nouvelles fonctionnalités pour faciliter son intégration dans des environnements complexes de production tel que les clusters. Vous pouvez consulter les nouvelles fonctionnalités de cette version en suivant le lien suivant :

https://www.ibm.com/developerworks/mydeveloperworks/blogs/fredho66/date/201204?lang=en_us

Informix Warehouse vous permet de lancer vos requêtes à partir des outils tel que COGNOS , Business Objects, Microstrategy, Pentaho sans aucune modification mais avec surtout un gain phénoménal en terme de performance. Plusieurs comparaisons ont été réalisées avec des outils comme Teradata, SQL Server, ou des produits de bases de données orientés en mode colonne comme Greenplum ou Vectorwise, et Informix Warehouse reste au-dessus du lot en termes de performance.

N'hésitez pas à tester ce produit fabuleux. Vous serez surpris.

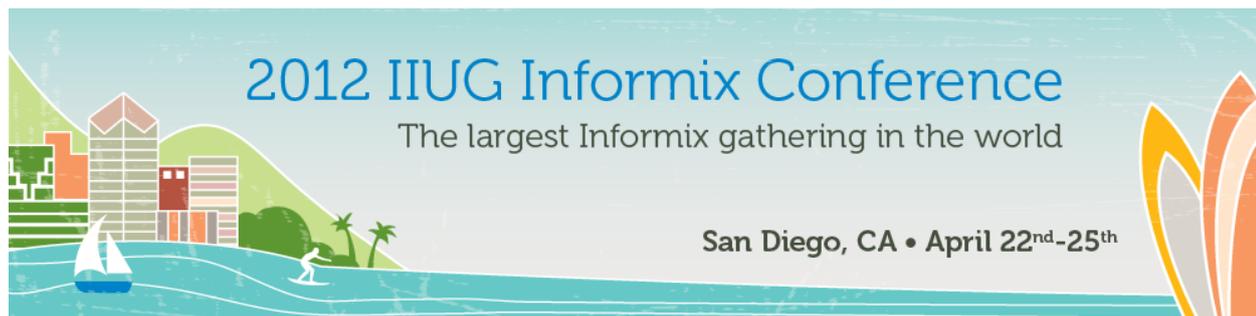
Nous sommes heureux de vous envoyer le second numéro de la Newsletter du User Group Informix France 2012.

Bonne Lecture et à très bientôt !

Khaled Bentebal - *Président de l'UGIF*
Olivier Bourdin - *Editeur de la Newsletter de l'UGIF*

Evénement : Retour sur l'IIUG à San Diego, Californie.

Du 22 au 25 avril 2012 a eut lieu la conférence **annuelle de l'International Informix Users Group (IIUG)** : <http://www.iiug.org/conf/2012/iiug/>



La conférence annuelle de l'IIUG a eu lieu à San Diego en Californie après plusieurs années à Overland Park – Kansas.

Ce fut la meilleure conférence depuis la création de l'IIUG.

Non seulement, il y a eu plus de participants et surtout d'utilisateurs et clients à la fois des Etats-Unis, d'Europe, d'Asie, d'Afrique, d'Australie et de Nouvelle Zélande, mais aussi plus de tracks (7 au lieu de 5 auparavant) et 123 sessions à la fois techniques et business par rapport à moins d'une centaine l'année précédente.

L'hôtel Marriott où eu lieu la conférence était complet plusieurs semaines avant la conférence, l'IIUG était obligé de prendre un deuxième hôtel pas loin pour pouvoir accommoder les participants venus de loin pour la plupart.

La conférence a commencé le dimanche 22 avril par une journée de tutoriaux.

Du lundi 23 au mercredi 25 et ce, pendant 3 jours pleins, les utilisateurs ont pu participé à 5 tracks techniques autour du moteur Informix, de l'offre Warehouse et diverses produits de l'offre Informix. Il y a eu pour la première fois un track « Business », un track spécialisé sur GENERO, et un track nommé « Hands on Labs » qui a permis à plusieurs utilisateurs d'expérimenter les sujets suivants :

- Informix Warehouse
- La sécurité avec Guardium
- Nouvelles fonctionnalités de la version 11.70 du moteur Informix
- TimesSeries
- Storage Optimization dans un environnement ER (Enterprise Replication)
- Flexible Grid (réseau hétérogène de système d'exploitation avec Informix)
- Optim et Informix

En parallèle, les utilisateurs ont pu passer gracieusement les certifications sur les produits Informix.

C'est le meilleur endroit au monde pour apprendre et développer ses compétences et enrichir ses connaissances sur les produits Informix.

Technical Tip: Nouvelles fonctionnalités en 11.70.xC5

Auteur : Olivier Bourdin

Fin juin 2012, la dernière version Interim 11.70.xC5 est sortie avec de nouvelles fonctionnalités. Voici une liste de ces nouvelles améliorations:

Administration

- Possibilité de planifier des réponses aux Event Alarms de faibles ou moyennes gravités.
- Nouvelle fonctionnalité IFX_BATCHEDREAD_INDEX.

Développement d'applications

- Amélioration de l'analyseur de mot-clé dans la Basic Text Search Datablade.
- Suppression de la restriction de la longueur maximale d'instructions SQL
- Performances réseau améliorées des requêtes
- Exemple de programme pour Change Data Capture (CDC)

Entreprise Replication

- Gestion des erreurs de réplication sur des leaf nodes

Global Language Support

- Nouvelle fonction `ifx_gl_complen()`

TimeSeries

- Compter les éléments TimeSeries qui matchent à des critères d'expression
- Supprimez les données TimeSeries périmées des containers
- Nouveaux opérateurs d'agrégation dans les données TimeSeries

▪ Nouvelle fonctionnalité IFX_BATCHEDREAD_INDEX

En utilisant l'option d'environnement IFX_BATCHEDREAD_INDEX par la déclaration SET ENVIRONNEMENT il est possible de contrôler l'optimiseur pour récupérer automatiquement un jeu de clés à partir d'un Index buffer pour une session. L'option active ou désactive la valeur du paramètre de configuration BATCHEDREAD_INDEX pour une session.

▪ Amélioration de l'analyseur de mot-clé dans la Basic Text Search Datablade

Vous pouvez spécifier que le keyword analyser élimine les espaces de fin de saisie de texte et de requêtes de sorte que vous n'avez pas besoin de spécifier le nombre exact des espaces pour interroger les mots dans les types de données de longueur variable. Pour cela il suffit d'ajouter le suffixe `.rt` dans le nom de l'analyseur mot-clé lorsque vous créez l'index `bts`.

▪ Suppression de la restriction de la longueur maximale d'instructions SQL

Auparavant, la taille des instructions SQL était limitée à 64 Ko. La longueur maximale d'instructions SQL et SPL routines est désormais de 4 Go. La seule exception est la longueur de l'instruction CREATE VIEW, qui est limitée à 2 Mo de longueur. La longueur étendue est applicable lorsque vous utilisez le client SDK 3.70.xC5 et JDBC 3.70.xC5.

- **Performances réseau améliorées des requêtes**

Vous pouvez maintenant définir la taille maximale du fetch buffer (mémoire tampon d'extraction) à 2 Go pour augmenter les performances des requêtes et réduire le trafic réseau. Pour cela, utilisez la variable d'environnement FET_BUF_SIZE. Plus grande est la taille du buffer, plus de rows peuvent être retournées. Dans ce cas, l'application cliente attendra moins fréquemment que le serveur de base de données renvoie les lignes. Un grand buffer peut améliorer les performances en réduisant l'overhead de remplissage du buffer côté client.

- **Exemple de programme pour Change Data Capture (CDC)**

Le programme d'exemple d'utilisation de l'API Change Data Capture, cdcapi.ec, est inclus dans le répertoire \$INFORMIXDIR/demo/cdc. Dans les versions précédentes, vous deviez copier le programme à partir de la documentation.

- **Gestion des erreurs de réplication sur des « leaf servers »**

Parce que les leaf serveurs ont des informations limitées sur les autres serveurs de réplication, les tables syscdrerror sur les leaf nodes ne contiennent plus les erreurs de réplication pour les autres serveurs de réplication.

- **Nouvelle fonction ifx_gl_complen()**

La fonction ifx_gl_complen() scanne plus rapidement les chaînes d'entrée qu'en utilisant la seule fonction ifx_gl_mblen(). La fonction ifx_gl_complen() renvoie la longueur en octets de la partie initiale d'une chaîne d'entrée qui correspond à un « collating element », ou renvoie 0 si la première partie de la chaîne n'est pas une « collation sequence ».

- **Compter les éléments TimeSeries qui matchent à des critères d'expression**

Vous pouvez compter le nombre d'éléments dans un TimeSeries qui correspondent aux critères d'une expression arithmétique simple en exécutant la fonction countlf(). Par exemple, vous pouvez compter le nombre d'éléments nuls.

- **Supprimez les données TimeSeries périmées des containers**

Vous pouvez supprimer les plus vieilles données TimeSeries à travers une date de fin dans un ou plusieurs conteneurs utilisés par plusieurs instances de TimeSeries en exécutant la fonction TSContainerPurge(). Vous pouvez ensuite réutiliser l'espace pour de nouvelles données.

- **Nouveaux opérateurs d'agrégation dans les données TimeSeries**

Vous pouvez retourner les éléments premiers ou les derniers entrés dans la base de données pour chaque timepoint en utilisant les opérateurs FIRST ou LAST dans la fonction TSRollup().

Technical Tip: Nouvelle fonctionnalité IFX_BATCHEDREAD_INDEX

Auteur : Yoram Benchetrit

Depuis Informix 11.50.xC6 l'interface des batched read a été introduite afin d'améliorer de manière significative les performances de tous les types de scan, et en particulier les performances des lights scans à l'aide du paramètre onconfig BATCHEDREAD_TABLE (cf. article sur les Light-Scan de la Newsletter du 1er trimestre 2010).

L'interface des batched read, est utilisée par défaut en version 11.70 et est contrôlée par les paramètres onconfig BATCHEDREAD_TABLE et BATCHEDREAD_INDEX. Cette technologie était originellement implémentée dans Informix XPS.

Voici la définition de ces paramètres dans le fichier onconfig :

```
#BATCHEDREAD_TABLE - Turn on/off xps api for table scans.  
#BATCHEDREAD_INDEX - Turn on/off xps api for index scans.  
BATCHEDREAD_TABLE 1  
BATCHEDREAD_INDEX 1
```

Le paramètre onconfig BATCHEDREAD_INDEX permet à l'optimiseur de lire simultanément un ensemble de clé d'index depuis un buffer de batch. Cette méthode de lecture des clés d'index permet d'améliorer les performances car les clés ne sont plus lues de manière séquentielle.

Depuis IDS 11.70.xC5 il est possible d'utiliser la variable d'environnement au niveau session IFX_BATCHEDREAD_INDEX de la commande SQL SET ENVIRONMENT, pour activer au niveau session la lecture automatique d'un ensemble de clés depuis un buffer d'index. La variable d'environnement IFX_BATCHEDREAD_INDEX active ou désactive la valeur du paramètre BATCHEDREAD_INDEX au niveau session.

Les valeurs possibles pour IFX_BATCHEDREAD_INDEX sont 0 ou 1.

1: permet à l'optimiseur d'activer au niveau session les fetchs automatique d'un ensemble de clés d'index depuis un buffer d'index

0: permet à l'optimiseur de désactiver au niveau session les fetchs automatique de clés d'index depuis un buffer d'index

Exemple:

Pour permettre à l'optimiseur de fetcher de manière automatique un ensemble de clés d'index depuis un buffer d'index pour une session, il s'agit d'utiliser la commande SQL suivante :

```
SET ENVIRONMENT IFX_BATCHEDREAD_INDEX '1';
```

Cette nouvelle commande SQL « *set environment* » peut être utilisée dans les procédures *sysdbopen()* afin que cette nouvelle fonctionnalité soit incorporée sans changement dans les codes applicatifs.

Technical Tip: Enterprise Replication – Pourquoi l'utiliser ?

Auteur : Laurent Revel

Depuis la version d'INFORMIX-Online Dynamic Server 7.21 (1996), Informix a intégré la réplication ER au cœur de son SGBDR. Pour rappel, cette fonctionnalité permet de répliquer un enregistrement (ou certains champs) sur N instances en mode synchro ou non et cela quelque soit la localisation des N instances. Ces dernières années, IBM a compris l'avantage de cette solution et travaille à faciliter son déploiement et administration.

Voici les principales raisons pour lesquels il faut utiliser cette fonctionnalité majeure d'IDS :

Mise en œuvre

- Après la configuration classique des hosts (/etc/hosts), de la synchronisation des machines sur la même heure (NTP), des règles de sécurité (host.equiv,...), et puis des 5 clics dans Open Admin Tool (le dernier GUI d'administration d'Informix) – une base de données sera répliquée.
- La contrainte de la clé primaire ayant été ôtée en version 11.70, toutes les bases de données peuvent donc maintenant être répliquées.

Administration

- Gérer les incidents, créer / modifier un nouveau réplicat (= définition pour Informix de la règle de réplication entre les instances et bases/tables), vérifier les conflits, resynchroniser des tables ... la plupart de ces tâches peuvent maintenant être faites via OAT.
- Des facilités de resynchronisation sont apparues ces dernières années, simplifiant la tâche du DBA lorsqu'un conflit arrive.

Développement

- Après avoir compris et étudié les règles de réplifications qui s'appliquent le mieux à vos besoins, très peu de développement (j'entends par là, de modifications du schéma de votre base de données) sont nécessaires. Seules certaines tables pourront avoir à être modifiées, si par exemple vous souhaitez avoir de la réplication actif – actif (auquel cas, vos tables devront être créées / modifiées en y ajoutant l'option WITH CRCOLS).

Monitoring

- Open Admin Tool s'est doté des premiers outils de monitoring. Même si dans un premier temps, OAT s'avère être un « simple portage » des commandes IER spécifiques de l'utilitaire onstat. Cela a au moins le bénéfice de ne pas avoir à se remémorer ces commandes peu usuelles et assez rébarbatives (onstat – g cdr).

Plusieurs clients français ont mis en œuvre cette fonctionnalité avec différents choix / complexité d'implémentations. Certains utilisent depuis de nombreuses années l'ER dans leur système de d'information, pour par exemple répliquer une production, d'autres pour répliquer des références de prix.

Et vous, y avez vous pensé – c'est finalement si simple ?

Technical Tip: Les applications Informix-4GL ont-elles un avenir ?

Auteur : Eric Verchelletto

Mais que faire avec ces applications Informix 4GL ?

Si vous arrivez sur cet article, il est une forte probabilité que cette question résonne à ce moment précis dans votre tête! Vous, ou vos prédécesseurs à ce poste, ont bien connu ce langage qui, sorti sur le marché au début des années 80, révolutionna à l'époque la manière de développer des applications de gestion sur la plateforme pour alors en pleine ébullition, à savoir Unix.

En 1984, Informix Software lance Informix-4GL, langage révolutionnaire construit à partir du langage C, mais qui contient des bibliothèques de fonctions suffisamment puissantes et synthétiques pour pouvoir développer rapidement des applications de gestion de type interactif. L'idée de génie ? Pouvoir « dessiner » des formes d'écrans d'un côté et écrire le code source interagissant avec la forme d'écran d'un autre côté.

A l'époque où j'écris cet article (avril 2012), force est de constater que les technologies de développement n'ont pas véritablement fait de progrès sur ce point. Certes, de beaux générateurs d'applications permettent de développer très vite quand il s'agit de fonctionnalités qui rentrent dans le moule standard des dits systèmes de développement.

Mais dès qu'il faut s'échapper un peu de ce standard, les difficultés surgissent vite, et l'on sombre avec grande facilité dans des usines à gaz inextricables. Très gourmandes en ressources-système, peu performantes, elles sont généralement fragiles mais trop souvent illisibles pour le pauvre collègue qui devra maintenir l'application quand vous serez en vacances. Y a-t-il vraiment progrès dans ces conditions ?

4GL, c'est quand même pas mal, et puis on n'a pas d'histoires avec !

Ce qui frappe précisément avec Informix-4GL, c'est la lisibilité du code qui influe directement sur la facilité et la rapidité de maintenance des applications, et ceci que vous soyez un expert aguerri aux tempes « presque » grisonnantes ou bien un jeune développeur débordant d'énergie. Ce que l'on voit généralement moins facilement (mais certains s'en chargent pour vous), c'est le côté performance et faible coût en ressources-système qui pourtant sont bien au rendez-vous.

Nonobstant, et toutes considérations rétrogrades que j'assume totalement dorénavant écartées, le fait que votre application développée en 1993 sur Informix-4GL et qui s'est enrichie au jour le jour jusqu'à maintenant ait pris des rides est indéniable. Mais elle est toujours là, elle ronronne et répond quotidiennement à vos besoins et exigences.

D'où le dilemme qui se pose entre garder une application ridiculement primitive que l'on a honte de montrer à toute personne née après 1970, et tout refaire à partir de zéro avec des outils plus actuels, de façon à ce que l'application soit « jolie » et adaptée au monde actuel.

Définition de la problématique et des hypothèses de travail.

A moins que l'activité de votre entreprise ait radicalement changé depuis vos dernières modifications, et que toute l'équipe de développement ait disparu, l'investissement fait par l'entreprise sur cette application est lourd, très lourd.

Prenons maintenant l'hypothèse « on met l'application à la poubelle et on refait tout ». Surgissent alors les propositions initiales :

Solution 0: On achète une application du marché faite pour notre métier, et on l'adapte/fait adapter par un intégrateur spécialisé.

J'ai travaillé pendant 8 ans chez puis pour un éditeur de ce genre d'applications, pourtant très réputé sur le marché. Mes conclusions ? Des produits très chers en termes de licences, et d'énormes dérives temporelles pour ce qui est de l'adaptation à l'entreprise cliente. J'ai constaté une grande proportion de projets abandonnés faute d'analyse fonctionnelle pertinente et d'existence d'une vraie analyse différentielle. La troisième cause est le triplement ou le quadruplement du budget soit parce qu'il y a eu un hors-sujet quant à la véritable vocation de l'application lors de l'acquisition, soit parce que mis à part le chef de projet, la majorité des développeurs apprennent à utiliser le système de développement à vos frais. A noter également que certains intégrateurs n'hésiteront pas à pousser ce genre de solution tout simplement parce qu'ils n'hésitent pas à faire passer leurs intérêts financiers avant vos besoins et votre satisfaction.

Solution 1 : On migre la base de données chez un autre éditeur et on refait l'application 4GL avec des outils plus récents qui marchent sur la nouvelle base de données?

Décision stupide n'ayant pour effets principaux que de rendre la concurrence heureuse et vous faire dire dans quelques mois : finalement avec les bases Informix on n'avait vraiment pas autant de soucis, que ce soit au point de vue besoin en personnel DBA, en stabilité et aussi en performance. Sachez à ce sujet que les retournements de projets vers Informix sont nombreux. Pour ce qui est des risques au niveau de l'application, voir solution n° 2. Si toutefois telle est votre décision, pour pouvez terminer votre lecture ici, et on se retrouve sans doute dans quelques temps.

Solution 2: On garde les bases Informix (sage décision), mais on se débarrasse de 4GL, parce que l'interface est dépassée et que les jeunes utilisateurs n'en veulent pas, et de plus il nous faut absolument une interface Web ?

Les arguments sont solides et acceptables, mais la contrepartie est lourde, comme l'explique le paragraphe précédent. De plus, l'application marche bien, peu (pas?) de bugs, et les utilisateurs la manipulent avec dextérité, efficacité et rapidité. Tout changer ne peut qu'engendrer un nombre important de risques nouveaux, tels que:

- risque de perte de qualité de l'analyse fonctionnelle, généré par la probabilité que cette analyse soit menée par des gens qui ne connaissent pas votre entreprise et ses rouages, ou, si c'est l'équipe 4GL qui mène le projet, il faut un temps d'adaptation non négligeable au nouvel outil de développement, et au-delà de la simple connaissance technique, on ne recapitalise pas vos 20 ans d'expérience acquis avec Informix-4GL en l'espace de 6 mois et même un an...
- simple risque de dérapage budget/temps du projet parce que ce risque est une constante dans tous les projets informatiques
- La logique de l'application est vraisemblablement différente et qu'il faut un temps d'adaptation pour les utilisateurs : ce point est inévitable. Ce fait, qui tient de la crainte du changement, est inévitable.
- Le risque de perte d'adhésion de la communauté des utilisateurs pendant un temps indéterminé, parce que l'application 4GL marchait bien, elle était plus simple et plus rapide à manipuler.

Donc vous pouvez atteindre vos objectifs avec la solution 2, mais le capital-risque est très important et il faut véritablement mettre les ressources adaptées en face.

Solution 3 : Et si l'on pouvait reprendre la quasi-intégralité de notre application 4GL, tout en lui offrant une interface graphique digne de ce nom ?

Si vous êtes totalement hermétique au 4GL, et que vous préférez d'autres environnements parce qu'ils sont plus robustes, plus performants en production, plus rapides en développement et plus faciles à maintenir, votre lecture peut s'arrêter ici, nous n'avons pas de solution pour vous.

Par contre si vous avez véritablement soupesé les pour et les contre de la réécriture d'une application nouvelle, vous arriverez sans doute à la conclusion que si l'on pouvait offrir une interface graphique à l'application 4GL sans trop d'efforts ni devoir apprendre un nouveau langage à partir de zéro, c'est plutôt là que se trouverait la solution la moins risquée et la plus rapide.

Depuis longtemps le besoin d'évolution se faisait ressentir

Depuis environ 1996, des sociétés tierces ont compris qu'Informix Software était en train de laisser un énorme vide quant à la demande des utilisateurs: le manque d'une interface graphique consistant pouvant fonctionner facilement sur un PC au lieu d'un terminal (ou émulation) vt100 et consorts.

Deux acteurs ont commencé à occuper ce terrain: une société à l'origine française (FourJS) qui avait compris l'opportunité que représentait un tel vide à combler. Même scénario pour une société britannique (Querix). Ces deux sociétés ont misé sur le concept de conserver le langage Informix 4GL, parce que c'est un langage qui a beaucoup d'adeptes et qu'il est productif, et de concevoir un compilateur/interpréteur permettant de faire fonctionner l'application en architecture trois-tiers, avec un client X-Windows ou plus tard un client Windows. Quelques années plus tard est venu s'adjoindre à l'aventure un projet open-source dont le nom est AUBIT-4GL.

Serait-ce à cause de la traversée du désert de la marque Informix consécutive au rachat par IBM, toujours est-il que ces « x4gl » ont apparemment évolué dans l'ombre pendant ces années, la tendance étant simplement à l'abandon total des produits Informix basé sur le fait qu'Informix était, à ce que disait la concurrence mais pas seulement, condamné définitivement.

C'était, on le sait maintenant, sans compter sur l'obstination des sociétés utilisatrices d'Informix à garder un produit qui donne totale satisfaction. Toujours est-il que ces produits sont toujours là et qu'ils commencent vraiment à monter en puissance, que ce soit au niveau des ventes qu'au niveau fonctionnalités.

Après l'acquisition par IBM, Informix 4GL comme prévisible a très peu évolué, mis à part quelques maintenance releases et quelques fonctionnalités nouvelles comme les Dynamic Arrays (enfin!) et la génération de WebServices. Heureusement, au fil des ans, les 3 produits de la famille « x4gl » ont perduré et se sont développés.

Les « x4gl » constituent aujourd'hui la meilleure réponse à votre besoin.

Partis de l'objectif initial se limitant au simple « revamping » des applications Informix 4GL, ces produits se sont progressivement transformés en de véritables studios de développement, enrichissant le langage Informix 4GL avec des fonctions qui manquaient crucialement à Informix 4GL, comme lire un fichier texte ou l'écrire sans passer par la fonction report (très puissante au demeurant et trop souvent délaissée au profit de code plus complexe). Les arrays dynamiques existent depuis un certain temps sur ces produits, alors qu'IBM ne les a introduits qu'en version 11.50. La manipulation des arrays a également été enrichie alors que 4GL n'évolue plus dans ce sens. On pourra toujours rétorquer qu'à

fonction manquante dans I-4GL, il est très facile de la développer en C, ce qui est vrai, mais cet argument rencontre aujourd'hui très peu d'adeptes.

Au-delà de l'aspect purement langage, l'énorme intérêt de ces solutions est la mise en oeuvre d'une architecture à trois composantes, (trois-tiers), que sont le serveur base de données, le serveur d'applications et les clients.

En séparant la couche de présentation des données à l'utilisateur de la couche de manipulation de ces données par la logique applicative, le client est allégé de toute la logique applicative, traitée dorénavant par le serveur d'applications qui la centralise. Le client se contente pour sa part de gérer l'interface-utilisateur, diminuant conséquemment la charge du PC par rapport à ce que rendrait une simple architecture client-serveur. On se retrouve donc dans une situation où le serveur est moins sollicité que dans une configuration avec des terminaux, et aussi où le client est moins sollicité car n'ayant pas à gérer la charge applicative. C'est ce qu'on appelle le 'thin client', 'light client' ou 'client léger'.

De là à se passer totalement du client, il n'y avait qu'un pas, qui a été franchi par tous les acteurs de la famille x4gl. La couche d'interface utilisateur étant séparée de la couche SQL et de la couche de logique applicative, il est facile de pouvoir adapter celle-ci de façon à la faire fonctionner à partir d'un navigateur internet, rendant alors possible le « zéro déploiement » ou la suppression de la nécessité d'installer quoique ce soit sur le client. Enorme avantage, notamment quand les clients sont répartis dans le monde.

Les versions les plus récentes de ces produits affichent également une intégration harmonieuse et parcimonieuse d'autres langages. C'est un fait que java peut faire des choses que 4GL ne fera jamais. Il est désormais possible d'intégrer des portions de code Java, invoquer des méthodes et déclarer des objets à partir de l'application 4GL, exactement comme vous appelleriez une fonction ou définiriez une variable en 4GL natif.

Il en est de même en ce qui concerne les générateurs de rapports. Au-delà du très puissant mais trop mode-texte « report » en 4GL, les éditeurs ont poussé le concept des rapports beaucoup plus loin, et surtout ont intégré une interface graphique et WYSIWYG, tels Genero Report Writer ou bien Lycia-BI de chez Querix. Au-delà de la simple conception avec un outil entièrement graphique, ces rapports peuvent aboutir à de véritables documents de Business Intelligence comparables aux grands noms du marché. Ils sont exécutés sur un «report engine» dédié, et peuvent être sauvegardés sous une grande diversité de formats (Excel, Word, pdf et beaucoup d'autres). Bien sûr, un ensemble de fonctions dédiées 4GL permet de manipuler ces rapports à partir de vos applications.

Dans les faits, ça me fait un projet de combien d'années ?

Une migration vers un x4gl se déroule généralement en deux phases :

Phase 1 : la compilation initiale.

Pour les programmes ne faisant pas appel à des commandes spécifiques au système sur lequel elles sont implémentées (cp, lp, cat, echo etc...), la compilation est généralement une affaire de minutes. Pour les programmes plus complexes, il faudra éventuellement intervenir dans le code pour résoudre ces questions d'appels-système si nécessaire. Suivant la taille de votre application, vous pouvez obtenir en une ou quelques journées une application qui fonctionnera dans l'environnement client que vous aurez choisi (Client Windows, Client Web, client mobile...).

L'application aura une interface « rudimentaire », mais pourra éventuellement être déployée si vous êtes pressé et que la communauté accepte temporairement une interface de transition. Elle inclura déjà l'utilisation de la souris, des boutons et quelques autres objets graphiques de base.

Phase 2 : la redéfinition avancée de l'interface graphique.

Le travail portera principalement sur la mise en forme des formes d'écran de façon à véritablement élaborer l'interface graphique. Vous pourrez définir votre charte graphique (couleurs, polices de caractères...) grâce au thème designer ou équivalent. Ce travail peut être un peu plus long que la simple compilation initiale, mais c'est là que sera la véritable valeur ajoutée du projet de migration. A noter que suivant l'éditeur, vous travaillerez plutôt, dans cette phase, sur le 4GL (Genero), ou plutôt sur les formes d'écrans (Querix Lycia), ce dernier ne requérant pratiquement aucune modification du code 4GL dans cette phase de mise en forme.

Libre à vous par la suite d'enrichir fonctionnellement l'application avec l'utilisation de blobs (images, documents Office, pdf, videos, etc...) et de donner à votre application l'élan dont elle avait besoin. Suivant la taille de votre application/le nombre de vos formes d'écran, cette phrase prendra au plus quelques semaines.

Conclusion :

Malgré tout ce que l'on raconte au sujet des « nouveaux » langages, ces derniers n'ont pas la flexibilité, la lisibilité, la stabilité et la performance que l'on est en droit de leur exiger en environnement de production. Ce point, discret au début du projet, se ressent au fil du temps de par le manquement des deadlines et l'augmentation incontrôlée des budgets. Malgré toutes les accusations de vétusté et de « has-been » proférées à son encontre, le 4GL n'en reste pas moins un véritable langage de programmation riche en possibilités et surtout redoutablement efficace en termes de productivité et de maintenance applicative.

En guise de cerise sur le gâteau, les éditeurs de la famille « x4gl » ont non seulement enrichi le langage 4GL pour l'amener au niveau fonctionnel des « nouveaux langages » permettant l'ouverture vers le multimédia, le web et le monde du « mobile », mais ils ont surtout réussi à en occulter la complexité de programmation, de façon à conserver la simplicité tellement appréciée et universellement acclamée d'Informix 4GL.

Alors, êtes-vous toujours prêt à abandonner une application qui marche pour la remplacer par une application dont vous ne savez quand elle sera prête à être réceptionnée et si elle sera acceptée par les utilisateurs ?

Derniers articles

- OUT/INOUT parameter support for SP calls in Informix ODBC Driver: [voir l'article](#)
- Certification exam 919 for Informix 11.70 prep, Part 2: Informix space management: [voir l'article](#)
- Optimizing Informix database access: [voir l'article](#)
- Get started now with the Informix TimeSeries solution: [voir l'article](#)
- Understand the trusted context feature in Informix: [voir l'article](#)

Vidéos

- JournalOfInformix [voir la vidéo](#)
- Interview with Stuart Litel (IIUG President): [voir la vidéo](#)
- IBM Informix growth strategies : An Interview with Bernie Spang: [voir la vidéo](#)
- Informix Warehouse Accelerator Accelerates Warehouse Queries up to 100X or More!: [voir la vidéo](#)
- IBM Informix Genero Accelerates a New Generation of Applications!: [voir la vidéo](#)
- Informix Dynamic Server for Tivoli Network Manager V3.9: [voir la vidéo](#)

Liens utiles

- Informix Developer Works : <http://www.ibm.com/developerworks/data/products/informix/>
- IBM Redbook : <http://www.redbooks.ibm.com/portals/data>
- IBM Data Management magazine: <http://ibmdatamag.com/>
- IIUG : <http://www.iiug.org/index.php>
- Informix sur facebook: <http://www.facebook.com/IBM.Informix>
- Informix sur Twitter : http://twitter.com/IBM_Informix
- The IIUG forums: <http://www.iiug.org/forums/technical.php>
- Blogs, Videos, News and more at : <http://planetids.com>
- Le channel sur Youtube de Jacques Roy : <http://www.youtube.com/user/jacquesroy58>
- Quick Reference Tool for Informix Business Partners : [accéder au site](#)
- IBM Software support lifecycle: [accéder au site](#)

Informix blogs

- Bruce Weed's blog <http://bruceweed.wordpress.com/>
- Fernando Nunes: Informix Technology: <http://informix-technology.blogspot.com/>
- Eric Vercelletto : Le village Informix <http://levillageinformix.blogspot.com/>
- Jacques Roy: <http://www.ibm.com/developerworks/blogs/page/jacquesroy>
- Informix blogs: <http://www.ibmdatabasemag.com/blog/main/archives/informix/index.html>
- The Informix Zone: <http://www.informix-zone.com>
- The Informix mag: <http://www.informixmag.com/>

Abonnement / Annulation / Avis

Cette newsletter est envoyée à des adresses enregistrées. Si vous souhaitez respectivement vous abonner ou vous désabonner, veuillez envoyer un mail avec pour sujet « ABONNER » ou « DESABONNER » à l'adresse email : ifmxnewsletter@fr.ibm.com.

Votre avis et vos contributions sont bien entendu les bienvenus ! N'hésitez pas à nous les faire parvenir à l'adresse email : ifmxnewsletter@fr.ibm.com.

Les contributeurs de ce numéro

Khaled Bentebal	Président du User Group InformixFrance Membre du board IIUG Directeur Général – Consultix
Eric Vercelletto	Directeur Général – BeGooden ITConsulting
Yoram Benchetrit	EMEA Informix L3 Advanced Problem Diagnostic IBM Software Group, Information Management
Olivier Bourdin	EMEA Informix L3 Advanced Problem Diagnostic IBM Certified Products Services IBM Software Group, Information Management
Laurent Revel	System X Benchmark Manager IBM Products & Solutions Support Center