

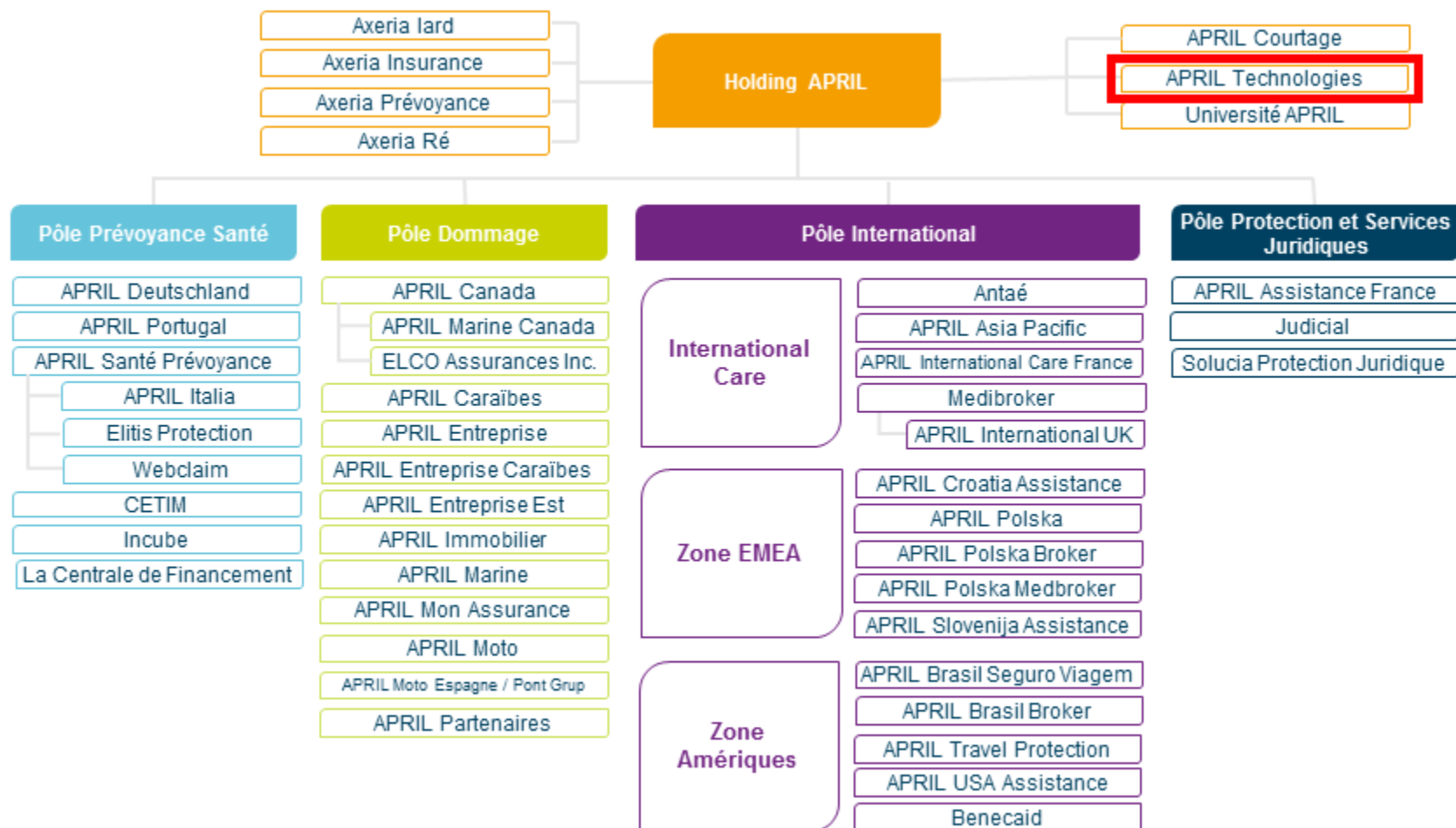
common

F R A N C E

Développements ILE







Contexte

- Développement de l'ERP maison
- Outillage maison
- Equipe de développement IBM i
 - \approx 30 personnes
- Une infrastructure déjà en place
 - Développement
 - Pré-production
 - Production

Besoin d'évolution

- Plus de SQL & ILE
 - Profitez des facilités offertes
 - Permettre des développements plus segmentés
 - Découpage BD / métier / IHM
 - Se rapprocher des concepts utilisés par les équipes .Net

- Limitations
 - Outillage maison
 - Ne sait pas gérer les concepts SQL (table, index, procédure ...) et ILE (module, programme de service ...)
 - En développement mais aussi en livraison

Les directions

- Mise en place d'outils
 - ALM
 - Déploiement
- En parallèle, travail sur les nouveaux standards de développement
 - ILE
 - SQL et carte F
- Formation des équipes
 - Outils
 - Langages

Objectifs

- Permettre un code
 - Plus lisible
 - Plus « architecturé »
 - Plus documenté
 - Plus outillé

La transformation

- Montée en compétence de l'équipe pilote
 - Qui deviendra le support interne des équipes sur
 - Le langage
 - Les outils
 - Les bonnes pratiques
- Implémentation des outils
 - Adaptation des outils aux particularités de l'entreprise
 - Personnalisation des cycles, des déploiements
 - Création d'outils de génération
 - En remplacement des outils

La transformation

- Recompilation de l'ensemble de l'application
 - Mise en place d'une politique de gestion des groupes d'activation
- Qualification / non régression

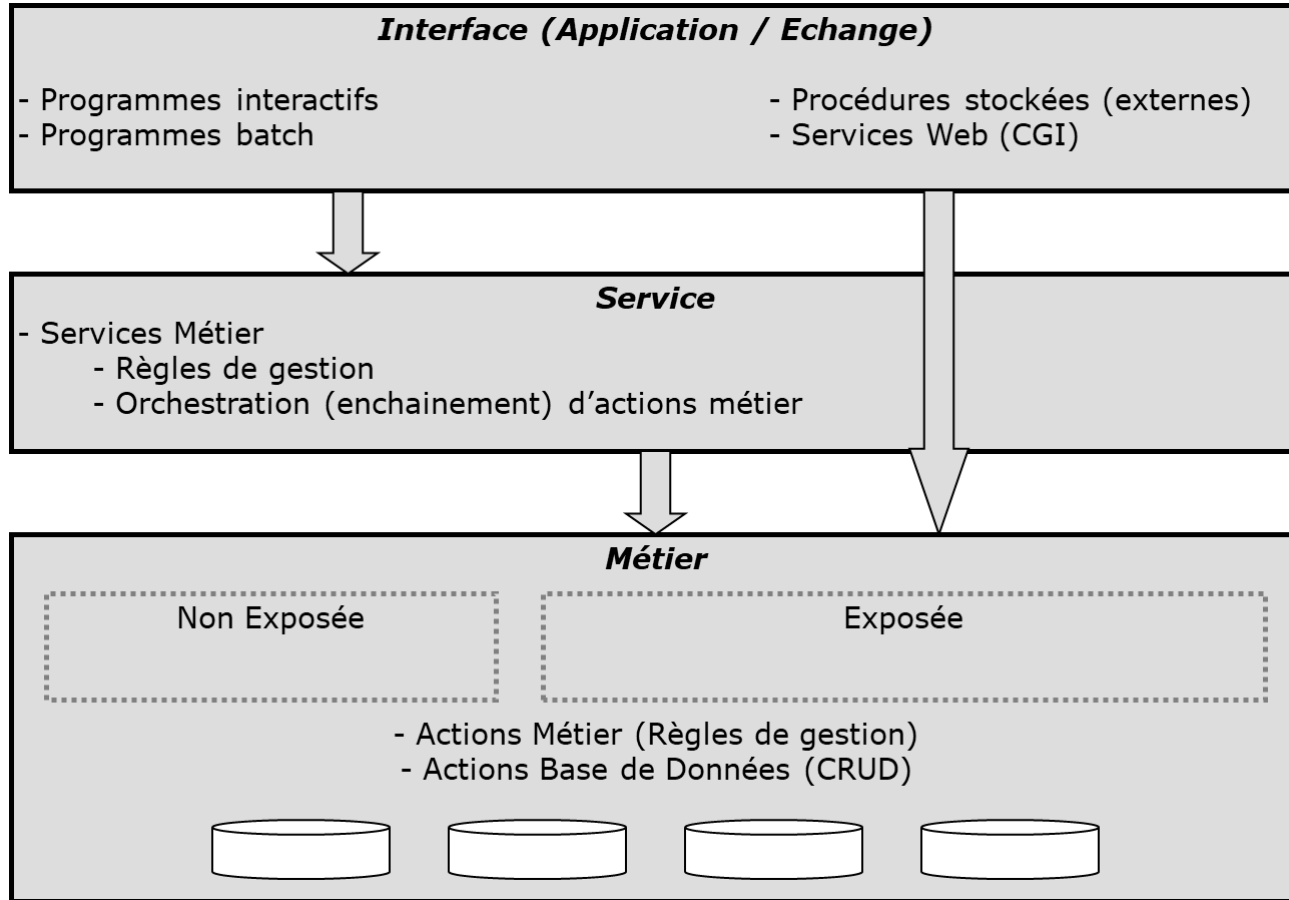
Impacts techniques principaux

- Portée des overrides (OVRDBF/OVRPRTF/OPNQRYF ...)
 - Par défaut *DFACTGRP
 - Si lancé depuis *DFACTPGRP (OPM) : *CALLLVL
 - Si lancé depuis un groupe d'activation : le groupe d'activation
- Consommation mémoire
 - En fonction de la gestion des groupes d'activation choisie
- Frontière de contrôle
 - Modification de la propagation des messages
- Programmes
 - RETURN/SETON LR
 - Fichiers en USROPN

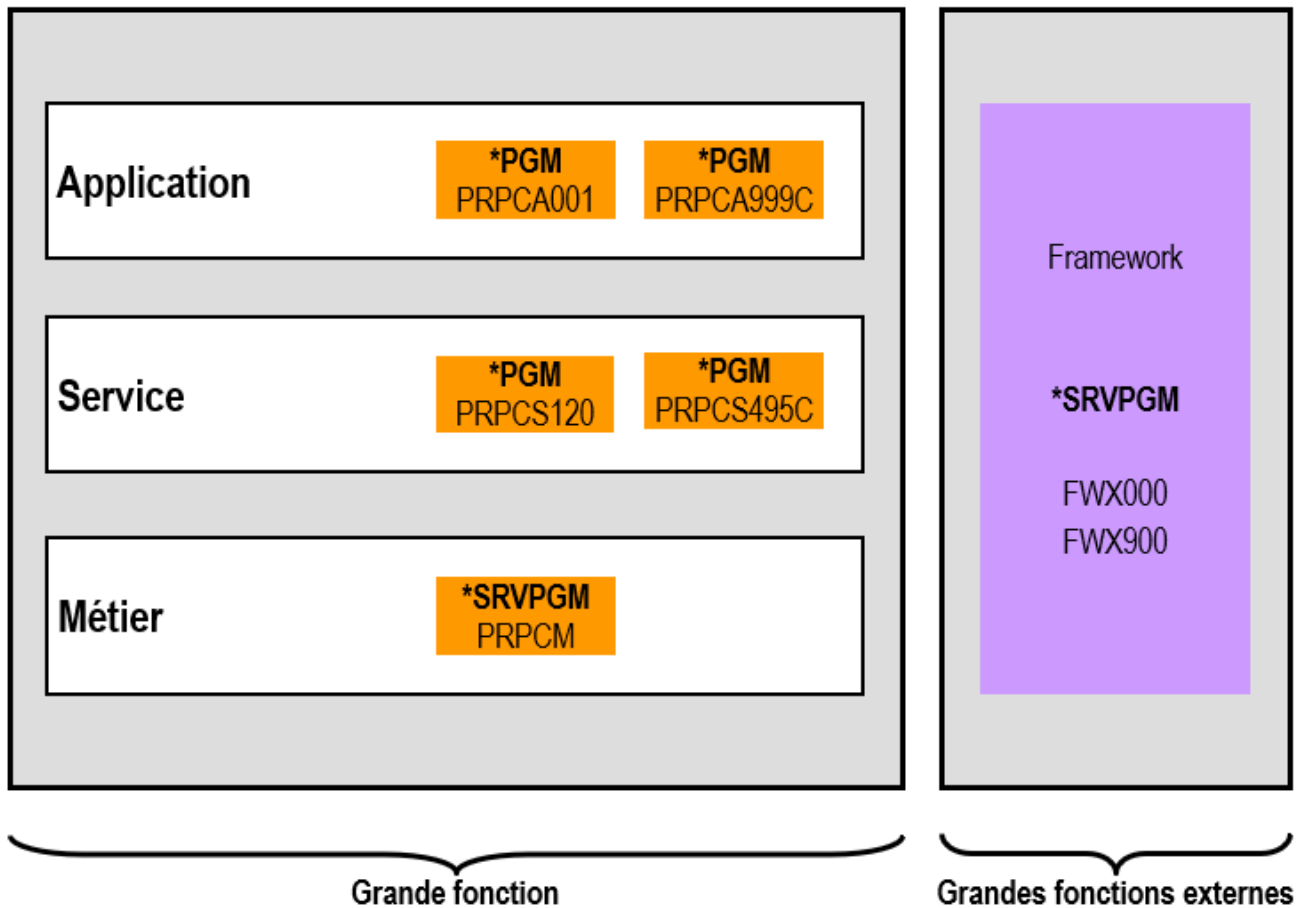
Architecture logicielle

- Cartographie du SI et urbanisation
 - Le nouveau code doit s'inscrire dans ce schéma
 - L'ancien code au fil des évolutions
 - Le code non modifié reste en l'état
- Définition d'un découpage technique
 - Calqué sur l'urbanisation
 - Permettant une indépendance BD / métier / interface
- Permet
 - De définir les dépendances techniques et fonctionnelles
 - De limiter les impacts dans les évolutions internes d'un « quartier »

Modèle



Modèle



Impact sur les équipes

- Une révolution dans la façon de coder
- Mais surtout de concevoir
 - Responsabilité du code vis-à-vis des appelants
- Ne pas sous-estimer le temps nécessaires pour la montée en compétence des équipes
 - Syntaxe
 - Conception
 - Culture
- Nécessité de pouvoir travailler en commun ou ménager des espaces d'échanges
- Ne pas aller trop loin dans la complexité technique ou l'abstraction du modèle
 - Perte d'efficacité en développement
 - Limitation de l'outillage
- Cela a permis de se mettre en position de profiter rapidement des évolutions disponibles sur le système
 - Web services, procédure/fonction SQL externes sur des programmes de services
 - Intégration de code Open Source (HTTPAPI, FTPAPI ...)
 - Meilleure intégration des langages (programme/programme de service constitués de modules de langages différents)
 - Intégration RPG/JAVA