

Power Week 2025

#pw2025

18 - 19 - 20 novembre 2025

IBM Innovation Studio Paris

S27 – Les Chemins de Modernisation

18 novembre 16:00 - 17:00

Gautier Dumas
CFD innovation
gdumas@cfid-innovation.fr

Pierre-Louis Berthoin
Gaia
plb@gaia.fr

IBM

common
FRANCE

Présentation

Pierre-Louis Berthoin

IBM i depuis 1989
Expert IBM i



GAIA / VOLUBIS

Formation (débutant, perfectionnement)
Expertise IBM i
Centre de Services



Présentation

Gautier Dumas

IBM i depuis 2009

Expert IBM i



CFD-Innovation

Formation (débutant, perfectionnement)

Expertise Web et Open Source IBM i

Centre de Services



Power Week

18 -19 - 20 novembre
2025



Les chemins de modernisation

Présentation

- Depuis plus de 20 ans, on entend parler de modernisation, c'est l'occasion pour nous d'expliquer avec du recul les grandes étapes d'une modernisation réussie.
- Le but de cette présentation n'est pas de faire une présentation technique, mais d'expliquer qu'il y a forcément un chemin pour vous et que la vraie erreur c'est de ne rien faire.
- On essaiera de vous donner des éléments simples de réflexion transposables à votre contexte

Premier Rappel

- La modernisation concerne plusieurs pends de votre applicatif
 - Le code
 - En majorité codée en RPG et en CLP
 - La base de données
 - En majorité Codée en DDS
 - L'IHM
 - En majorité des développements en 5250 à Base de DDS
- Que voulez moderniser et aussi pourquoi voulez vous le faire ?

Quelques conseils

- Le plus important va être de faire le bon choix du chemin à suivre.
- Vous devrez choisir la route qui est la plus adaptée à votre organisation en tenant compte à minima des éléments suivants :
 - Le nombre de personnes disponibles pour ce projet
 - Les connaissances internes et la capacité à transmettre
 - Les compétences présentes dans l'entreprise (Développeur PHP, Nodejs , .Net, etc ...)
 - Les obligations légales de votre secteur d'activité
 - Votre secteur géographique
 - Etc ...

Quelques conseils

- Il faut se lancer, vous n'êtes pas obligé de tout moderniser, vous pouvez faire des opérations au fil de l'eau
- N'oubliez pas
- « Un intellectuel assis va moins loin qu'un con qui marche ». Audiard

Le code

- L'enjeu majeur qu'on a constaté c'est la maintenance de ce code dans les années à venir.
- Les nouvelles ressources que nous appelons affectueusement les Kevin ne savent travailler qu'en Free RPG, et maîtrise VS code et SQL
- Vous avez ici les 3 mots clés qui vont vous donner une première indication de la cible
- Attention :
 - Moderniser ce qui sert dans votre application

Le code

- Pour nous la cible minimum à viser est le RPG IV
 - Elle n'est pas destructive, en effet cette migration ne casse pas l'existant et ne nécessite pas de grosses phases de test.
 - Elle permet d'ajouter du code en free directement dans le source du programme.
 - Vous rentrez dans le monde de l'ILE et vous bénéficiez de la possibilité d'intégrer des fonctions et des procédures
- Si vous pouvez convertir en Free c'est mieux, mais l'impact est plus important (GOTO, Cycle Gap, etc..)

Le code

- Quand doit on réécrire un programme ?
- Il est important de déterminer les programmes qui ne devront pas être migrés mais qui devront être réécrits.
 - Cycle Gap ?
 - Nombre de bugs corrigés sur année
 - % de Code inutilisé dans un programme (couverture de code ...)
 - La technologie utilisée (qui peut être remplacée par des fonctions plus récentes, exemple génération de fichiers csv).
 - Etc ...

Quelques chiffres

- La dette est énorme
- 7 % des sociétés ont encore du 36
 - Dont 3 % des programmes stratégiques
- 42 % du code est encore en RPG III
- 60 % des éléments base de données sont en DDS
- *Eléments collectés sur 102 clients GAIA ou Volubis, pas pris en compte les comptes COBOL

N'oubliez pas

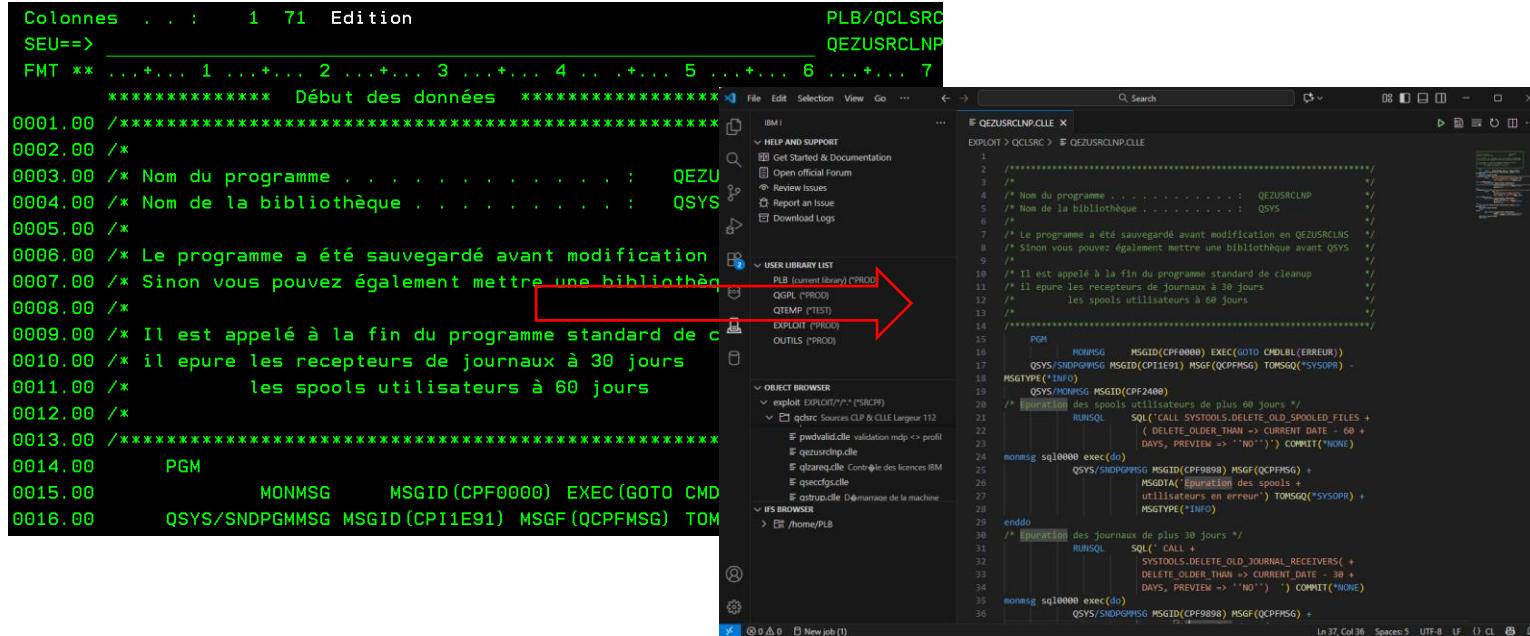
- Pour moderniser ou maintenir une application , vous devrez la connaître fonctionnellement:
- Le plus souvent, on a un manque de documentation énorme.
- On peut collecter les docs existantes, il en existe plus qu'on croit, mais mal référencées, et souvent pas à jour
- On peut interroger les sachants et produire ou mettre à jour certaines documentations
- Et après il y a l'IA qui peut vous aider ...

Du RPGII vers le RPG FREE



Les outils et bonnes pratiques

- Développement de SEU vers VScode.



```
Colonne 1 71 Edition PLB/QCLSRC
SEU==> QEZUSRCLNP
FMT ** ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Début des données *****
0001.00 /*****
0002.00 /*
0003.00 /* Nom du programme : QEZU
0004.00 /* Nom de la bibliothèque : QSYS
0005.00 /*
0006.00 /* Le programme a été sauvegardé avant modification
0007.00 /* Sinon vous pouvez également mettre une bibliothèque
0008.00 /*
0009.00 /* Il est appelé à la fin du programme standard de c
0010.00 /* il epure les recepteurs de journaux à 30 jours
0011.00 /* les spools utilisateurs à 60 jours
0012.00 /*
0013.00 /*****
0014.00 PGM
0015.00 MONMSG MSGID(CPF0000) EXEC(GOTO CMD
0016.00 QSYS/SNDPGMMSG MSGID(CPI1E91) MSGF(QCPFMSG) TOM
```

```
1
2 /*****
3 /*
4 /* Nom du programme : QEZUSRCLNP
5 /* Nom de la bibliothèque : QSYS
6 /*
7 /* Le programme a été sauvegardé avant modification en QEZUSRCLNP
8 /* Sinon vous pouvez également mettre une bibliothèque avant QSYS
9 /*
10 /* Il est appelé à la fin du programme standard de cleanup
11 /* Il epure les recepteurs de journaux à 30 jours
12 /* les spools utilisateurs à 60 jours
13 /*
14 /*****
15 PGM
16 MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(ERRURS))
17 QSYS/SNDPGMMSG MSGID(CPI1E91) MSGF(QCPFMSG) TOMSGQ('SYSOPR')
18 MSGTYPE(*INFO)
19 QSYS/MONMSG MSGID(CPF2400)
20 /* Epuration des spools utilisateurs de plus 60 jours */
21 RUNSQL SQL('CALL SYSTOOLS.DELETE_OLD_SPOOLED_FILES +
22 (DELETE_OLD_THAN => CURRENT_DATE - 60 +
23 DAYS, PREVIEW => 'NO')') COMMIT(*NONE)
24 monmsg sql0000 exec(do)
25 QSYS/SNDPGMMSG MSGID(CPF0000) MSGF(QCPFMSG) +
26 MSGDTA(' Epuration des spools +
27 utilisateurs en erreur') TOMSGQ('SYSOPR') +
28 MSGTYPE(*INFO)
29 enddo
30 /* Epuration des journaux de plus 30 jours */
31 RUNSQL SQL('CALL +
32 SYSTOOLS.DELETE_OLD_JOURNAL_RECEIVERS( +
33 DELETE_OLD_THAN => CURRENT_DATE - 30 +
34 DAYS, PREVIEW => 'NO')') COMMIT(*NONE)
35 monmsg sql0000 exec(do)
36 QSYS/SNDPGMMSG MSGID(CPF0000) MSGF(QCPFMSG) +
```

Les outils et bonnes pratiques

- Mise au point SQL, STRSQL vers ACS

```
Entrée d'instructions SQL

Saisissez l'instruction SQL, puis appuyez sur ENTREE.
==> select job_name, state_timestamp, user_name, t.*
      from qsys2.db_transaction_info t
      where local_changes_pending = 'YES'
      order by t.state_timestamp;
```

F3=Exit F4=Invite F6=Insérer ligne F9=Rappel F10=Copier
F12=Annuler F13=Services F24=Autre

Exécution de scripts SQL - Neptune(Neptune)

Fichier Edition Recherche Affichage Connexion Exécution Explication Moniteur Editeur Outils Aide

*Sans titre 1 *

```
1 select job_name, state_timestamp, user_name, t.*
2   from qsys2.db_transaction_info t
3   where local_changes_pending = 'YES'
4   order by t.state_timestamp;
```

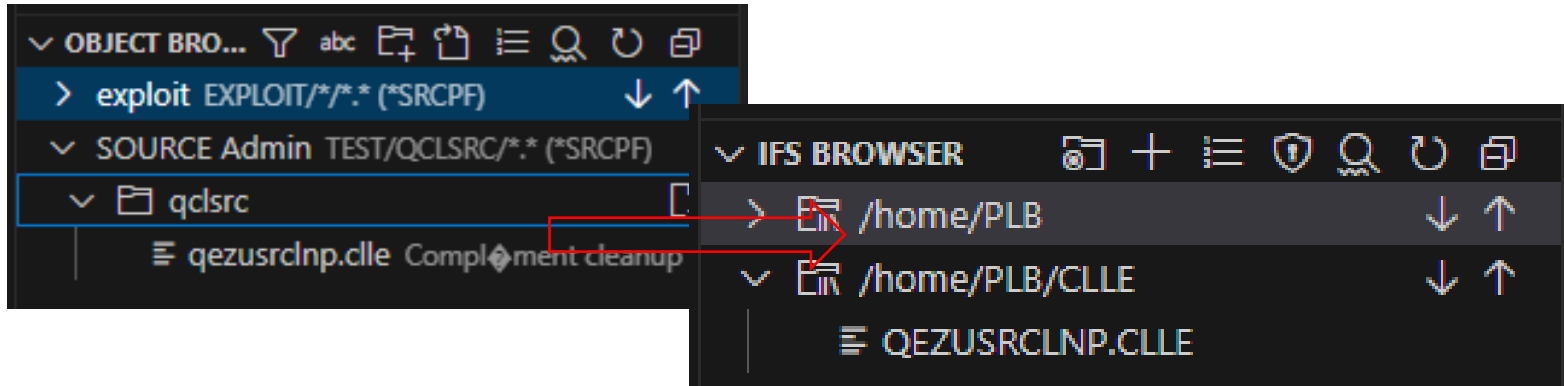
Job Name	State Timestamp	User Name	Commitment Definition	Commitment Definition Description	Commitment Definition ID	Job Name	TI
JOB_NAME	STATE_TIMESTAMP	USER_NAME	COMMITMENT_DEFINITION	COMMITMENT_DEFINITION_DESCRIPTION	COMMITMENT_DEFINITION_ID	JOB_NAME	TI

Terminé : 0 lignes extraites

Messages | Environnement | select job_name, state_timestamp, user_name, t.* from qsys2.db_transaction_info t where local_change...

Les outils et bonnes pratiques

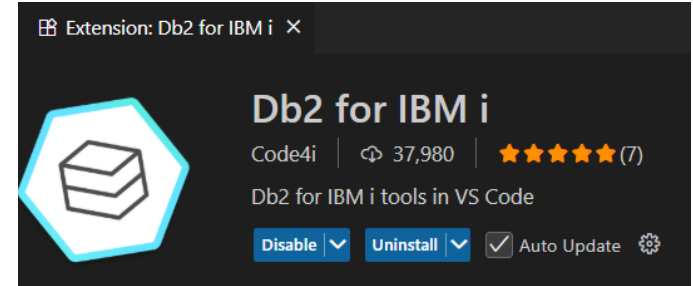
- Le système de fichiers, fichier sources vers IFS



- ** Facilite la mise en place de Git, mais rend inutilisable SEU

Les outils et bonnes pratiques

- VSCode et l'extension **Db2 for IBM i**
- Les fonctionnalités du module SQL d'ACS
 - Exécution SQL, vue schémas, historique des requêtes, accès aux exemples, Visual Explain ...
- Mais aussi
 - SQL prefixes (**json, csv, sql** et **rpg...**)
 - Mode édition (**update:**)
 - Graphiques avec SQL
 - Notebook
 - @db2i et SQL Code assistant (IA intégrée) !
 - <https://codefori.github.io/docs/extensions/db2i/ai/code-assistant/>
- <https://codefori.github.io/docs/extensions/db2i/>



Conclusion

- Une fois votre migration finie, vous devrez :
 - Continuer à faire évoluer vos pratiques en intégrant les nouvelles possibilités offertes par la plateforme, et les standards du marché.
 - Extraire les règles métier de vos applications pour les publier de manière dans vos nouvelles solutions, pour faire de l'APIsation par exemple
- C'est à ce prix que vous pérenniserez votre investissement, attirerez et garderez de nouvelles recrues qui assureront l'évolution de votre SI.

La base de données

- Vous pouvez facilement migrer des fichiers de PF à Table à 3 conditions:
 - Ne pas avoir de clé non unique, SQL ne pourra transformer cette clé en contrainte
 - Conserver le niveau de format de vos fichiers qui est contrôlé dans tous vos programmes
 - La qualité de vos données, SQL contrôlant à l'injection dans la base et non à la restitution

➔ Vous pourrez alors faire des accès par SQL qui seront plus claires et plus performants, attention à une petite chose, la nouvelle gestion des verrouillages ...

La base de données

- Même sans migrer vers du SQL
 - Vous pouvez mettre des noms long, grâce aux alias et sans perdre votre niveau de format
 - Vous pouvez changer la taille des pages d'index sur les LF pour améliorer les performances de l'optimiseur SQL
 - Vous pouvez matérialiser des relations DB , en créant des contraintes d'intégrité référentielle

Que dit la loi

- **Code de la consommation** – Article L.111-1 (Devoir d'information)
 - Si l'ESN travaille avec un client non professionnel, elle a une obligation d'information sur :
 - La compatibilité du code avec les évolutions technologiques,
 - Les risques d'obsolescence rapide.
 - 📌 Cas pratique :
 - Une ESN développant un logiciel en COBOL sans avertir le client des difficultés futures pourrait être mise en cause.
- **Code civil** – Article 1231-1 (Responsabilité contractuelle)
 - Si une ESN livre un logiciel rapidement obsolète, elle pourrait être tenue responsable si :
 - L'obsolescence était prévisible et non signalée,
 - Il y a un défaut de conseil sur la durée de vie du logiciel.
 - 📌 Conséquence :
 - Un client peut réclamer un dédommagement s'il prouve que l'ESN a sciemment utilisé des technologies obsolètes.

Architecture applicative

- Dans une application 5250, vous pouvez développer en respectant le modèle MVC , mais ce n'est pas le cas sur les applications existantes.
- Même sur un programme 5250, vous pouvez faire du code très lisible et maintenable en intégrant de L'ILE par exemple

L'IHM

- Vous pouvez améliorer les affichages 5250,
 - Paramétrage ACS
 - Ajout de mots clés graphique
- C'est une solution tactique qui permet de sortir du mode noir et vert

Plus de détails

Session 85 : Donnez une seconde vie aux écrans 5250

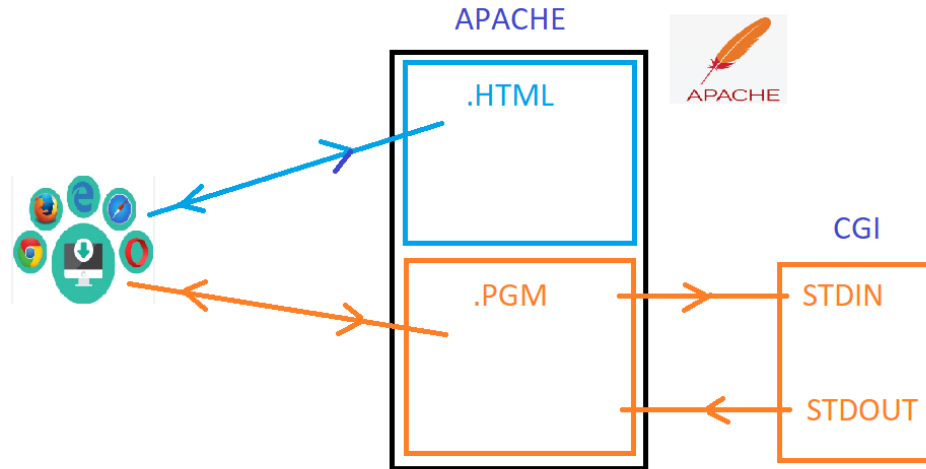
Jeudi 20 novembre – 15:00 - Tuileries

Florian Gradot & Pierre-Louis Berthoin

- Vous pouvez également utiliser une solution de rewamping plus ou moins impactante sur l'existant, le conseil qu'on vous donne c'est de bien faire un POC qui soit représentatif de ce vous voulez présenter

Et CGI

- Pour faire du web en RPGLE, dans une instance Apache



- Solution tactique qui permet de webiser rapidement une ou deux transactions sans techno externe

Administration et supervision IBM i

- La modernisation IBM i ne concerne pas que le patrimoine applicatif
- Il s'agit également d'intégrer la gestion du système dans la stratégie globale de l'IT de l'entreprise et de l'ouvrir au plus grand nombre
- Cela concerne par exemple
 - La simplification de l'accès à l'administration de l'IBM i
 - Utilisation de [Navigator for i](#)
 - Utilisation des [IBM i Services](#)
 - La mise en œuvre de solutions de supervision
 - À base de [Grafana](#), [Centreon](#) ... Ou autre solution du marché !
 - Des exporters ou système d'alerting en temps réel ([Manzan](#))
 - La mise en œuvre de logs ou de seuils
 - [SELF](#), [Query Supervisor](#)
- On peut commencer simplement avec de petites choses

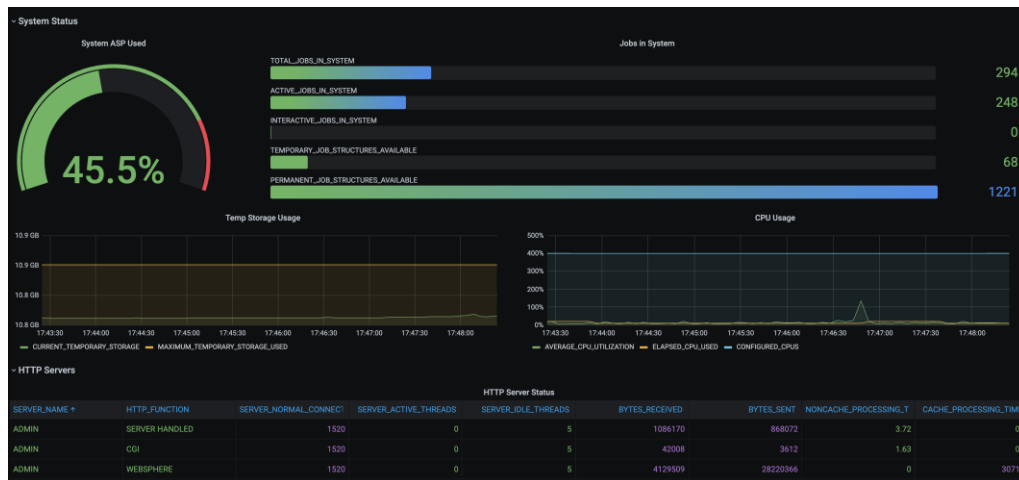
Administration et supervision – Exemples

- Manzan – Solution Open Source d'alerting en temps réel

Plus de détails

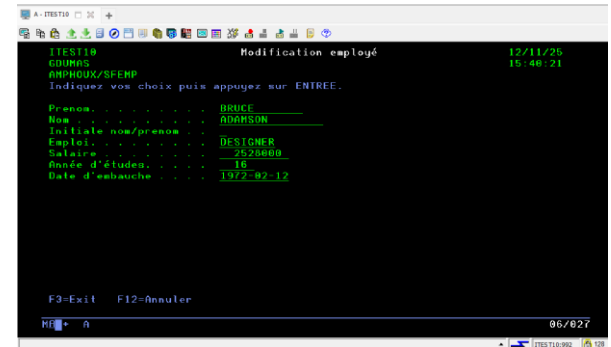
Session 66 : Manzan : vers une solution unifiée des évènements IBM i
Jeudi 20 novembre – 11:15 - Rivoli
Gautier Dumas

- Grafana + Exporters pour visualiser en live l'état de santé de l'IBM i



Pourquoi moderniser l'IHM ?

- L'IHM (Interface Homme Machine) est le pont entre l'humain et la machine
- C'est le premier contact d'un utilisateur avec le système en place !
- Il est important d'y apporter tout le soin nécessaire pour
 - Favoriser l'adoption d'une application par ses utilisateurs
 - Faciliter les interactions avec la machine
 - Eviter un premier ressenti négatif - *difficile à gommer ensuite*
- L'écran vert est souvent synonyme de "minitel"
 - Méconnu des nouveaux arrivants
 - Techniques (développeurs, admins ...)
 - Métiers (commerciaux, vendeurs, gestionnaires...)
 - Mais aussi des directions (DSI, direction générale ...)
 - Même si décrié parfois à tort
 - Bonnes performances
 - Saisie en masse



Modernisation IHM – Interfaces web

- Les interfaces web, oui ! Mais pour quels projets ?
- Avantages généraux :
 - Technologies standards connues du plus grand nombre
 - Développeurs comme utilisateurs
 - Rien à déployer sur les postes des utilisateurs
 - Centralisation des développements
- On peut distinguer plusieurs types de projets
 - Webisation et rewamping des écrans 5250
 - Identification et webisation de modules de l'ERP ciblés
 - Nouvelles applications répondant à des nouvelles demandes et besoins métier



Webisation et rewamping des écrans 5250

- Moderniser les écrans 5250 via une solution éditeur permet d'améliorer rapidement l'expérience utilisateur
 - Interface plus moderne, meilleure adoption, productivité accrue
 - Tout en facilitant un déploiement large sans réécriture des logiques métier
- Cependant, cette approche introduit une dépendance forte à l'outil choisi
 - Avec des coûts récurrents (licences, maintenance)
 - Des limitations fonctionnelles selon l'éditeur
 - Et la nécessité de disposer ou former des ressources compétentes sur la solution
- Et une modernisation superficielle possible

Webisation avec développement web

- Développer de nouvelles applications web sur IBM i offre une liberté totale de conception :
 - On peut repenser l'ergonomie, créer des interfaces modernes et faire évoluer séparément les couches front et back
- Cette approche permet d'introduire progressivement de nouvelles briques techniques (API, microservices, frameworks)
 - Tout en capitalisant sur les données, les programmes et la sécurité de l'IBM i
 - Ouvrant le SI vers d'autres systèmes en favorisant la collaboration avec ses partenaires / collaborateurs
 - Contribue à la montée en compétences des équipes autour des technologies web standards (HTTP, JSON, SQL, API...) offrant une meilleure attractivité

Webisation avec développement web

- L'investissement humain initial peut être plus important
 - Repenser l'architecture, concevoir l'UI, créer les API – plus de travail qu'un rewamping
 - Mais possible de commencer par des petits projets
 - Nécessite des compétences web pour
 - Le développement des applications
 - L'administration de ces nouveaux environnements
- Points de vigilance :
 - Gestion de la cohérence fonctionnelle entre le front et le back
 - Eviter l'empilement de micro-projets avec une gouvernance claire

Bonnes pratiques

- Quelque soit le projet, il est important :
 - De bien identifier les ressources et les connaissances déjà en place dans l'entreprise
 - De tester avec des POC dans vos environnements pour bien se rendre compte des possibilités et des contraintes
 - De cibler et d'impliquer des utilisateurs (dès le départ)
 - Premier déploiement dans un service pilote avant déploiement au reste de l'entreprise
 - D'identifier des utilisateurs clés qui vont faire le pont entre les usages métiers et les équipes de développements
 - Documenter les nouveaux flux et les nouvelles applications
 - Faciliter la communication entre les utilisateurs et les équipes de développement
 - Avoir une feuille de route claire sur les objectifs à atteindre

Modernisation IHM – Culture web

- Une culture du développement web
 - De nouveaux éditeurs de code (VSCode) devenus des standards
 - peu importe le langage de développement adopté dont RPG maintenant
 - Les API REST / SOAP
 - CI / CD => versioning git, intégration et déploiement continu
 - Une ouverture culturelle et technique
 - Au reste du SI
 - Aux interlocuteurs techniques externes
- Qui peut s'appliquer aux développements IBM i natif !

Exemples architecture des applications web IBM i

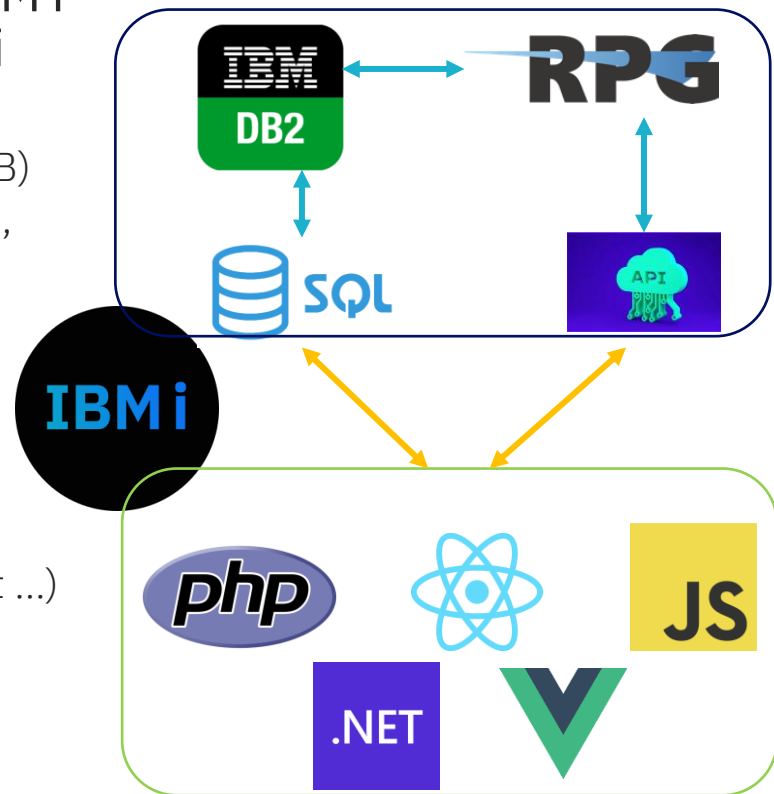
- Contexte 1 : Implication forte des équipes IBM i et conservation des règles métiers côté IBM i

- Côté Back :

- Programmes RPG / COBOL modernisés (cf. PLB)
- DB2 avec du DataCentric (contraintes, triggers, RCAC...)
- Exposition de ces règles en SQL (fonctions, procédures stockées ...)
- Exposition de ces règles par API Web Service

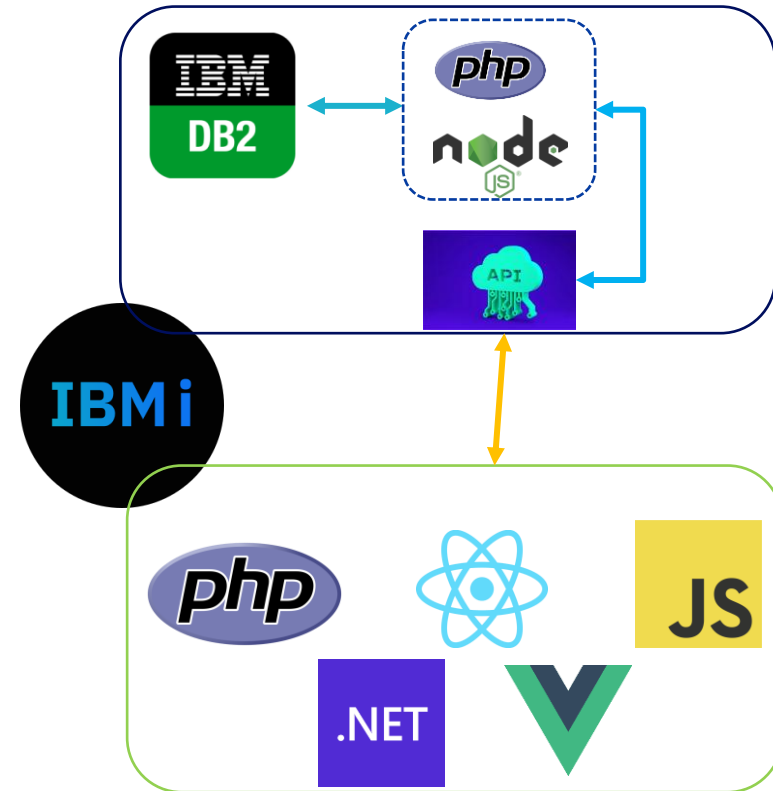
- Côté Front :

- Adoption d'une technologie Web (PHP, Js, .net ...)
- Accès aux règles métiers par :
 - SQL
 - API Web Services



Exemples architecture des applications web IBM i

- Contexte 2 : Nouveau projet / nouvelle équipe - Full Web
 - Côté Back :
 - PHP, Node pour accès aux données
 - API Rest
 - Côté Front :
 - Adoption d'une technologie Web (PHP, Js, .net ...)
 - Accès aux règles métiers par :
 - SQL
 - API Web Services

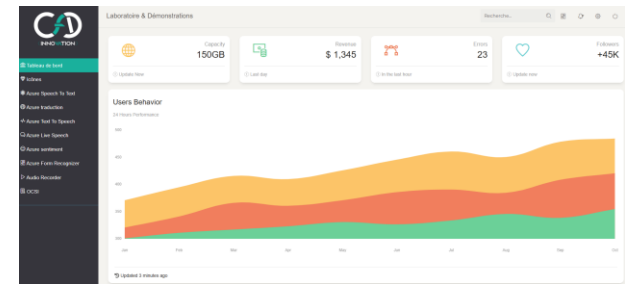


Framework or not Framework

- Encore une fois, la réponse à cette question dépend de votre contexte !
- Quelques points d'attention lors de l'adoption d'un framework :
 - S'assurer du degré de facilité des montées de version
 - S'assurer d'adopter une solution avec une communauté réelle (attention aux effets de mode)
 - L'utilisation des ORM souvent poussées dans les frameworks
 - Très pratique pour des applications from scratch
 - Parfois compliqué avec le legacy
- Ces facteurs doivent également être pris en compte
 - La typologie et la taille du projet
 - Déployer un framework pour un petit projet n'est pas forcément une bonne idée
 - La complexité est l'ennemie du bien
 - Les compétences disponibles
 - En interne
 - En externe

Exemples de projets sur IBM i

- Application Web pour les commerciaux / ADV
 - Suivi clients
 - Suivi des opportunités
 - Affichage des stocks
 - Gestion de produits améliorés
 - Ajout d'images
 - Ergonomie graphique et repensée
- Portail Web pour des analyses BI
 - Statistiques visuelles et graphiques
 - Export vers de la bureautique
- Edition de documents d'expédition PDF
 - avec intégration de QR Code
- Application mobile
 - Pour les inventaires
 - Pour des enregistrements d'entrée / sortie de stock par scan de code barres
- ...



Article scanné

Code article : 323601T

Quantité à sortir :

☒ Valider la sortie

Le choix !

- Il n'existe pas une bonne architecture pour tout le monde...
- ...il existe une bonne architecture pour un besoin et un contexte donné !
- La bonne nouvelle, c'est qu'il existe des solutions matures pour moderniser les différents volets de l'IBM i
 - L'existant technique, socle de toute construction applicative pérenne
 - on bâtit une maison sur des fondations solides !
 - Les nouvelles technologies pour répondre à toutes les demandes business
 - L'administration et la supervision de l'IBM i
- Il n'y a pas une solution unique, mais une solution qui est adaptée pour vous
 - Et ne rien faire...
 - ... C'est une erreur !
- Alors, on commence quand ?

MERC

