



Plateforme de traitement de données environnementales

Un projet *open source* pour les ingénieurs des données

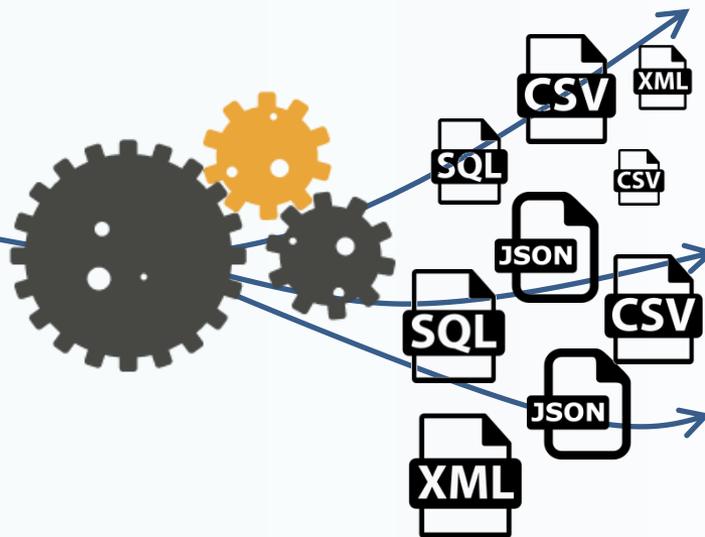
Alexandre Liccardi - ONEMA





Le projet (contexte & objectif)

Traiter les données environnementales



The screenshot shows the eaufrance website interface. At the top, it says "eaufrance Le service public d'information sur l'eau". Below that, there's a section for "data.eaufrance.fr" with a map and text: "Exemple d'utilisation de l'API H2O/EAU avec les données 'poissons'". Further down, there are logos for "cartograph" (Données de synthèse sur l'eau) and "NAIADES" (Données sur la qualité des eaux de surface). At the bottom, it lists "SNPE, SISPEA, BNPE... et bien d'autres".



Le projet (contexte & objectif)

Une boîte à outils

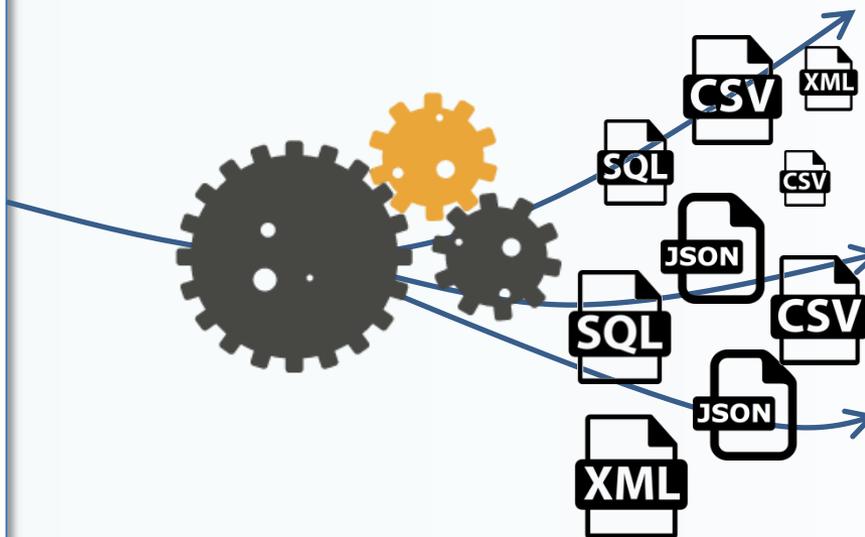
Calculs d'indicateurs d'**état des eaux**, avec règles d'agrégations et effets de seuil (SEEE, PAP...)

Données historiques de **qualité très hétérogènes**, de **formats différents**, fournies par des opérateurs différents (SEEE, flux Agences de l'Eau, CAMPEX...)

Correction à la volée de données, sur des erreurs connues, pour exploitation scientifique (PCB...)

Croisement d'échelles administratives, socio-économiques, physiques (SISPEA, BNVD...)

Reconstruction de données manquantes ou incohérentes (Rapportages DCE, Carthage...)



Jointures et requêtes paramétrables

Injection de code sur des sélections

Production automatisée de fichiers tabulaires et graphiques

...

Traçabilité des opérations complexes



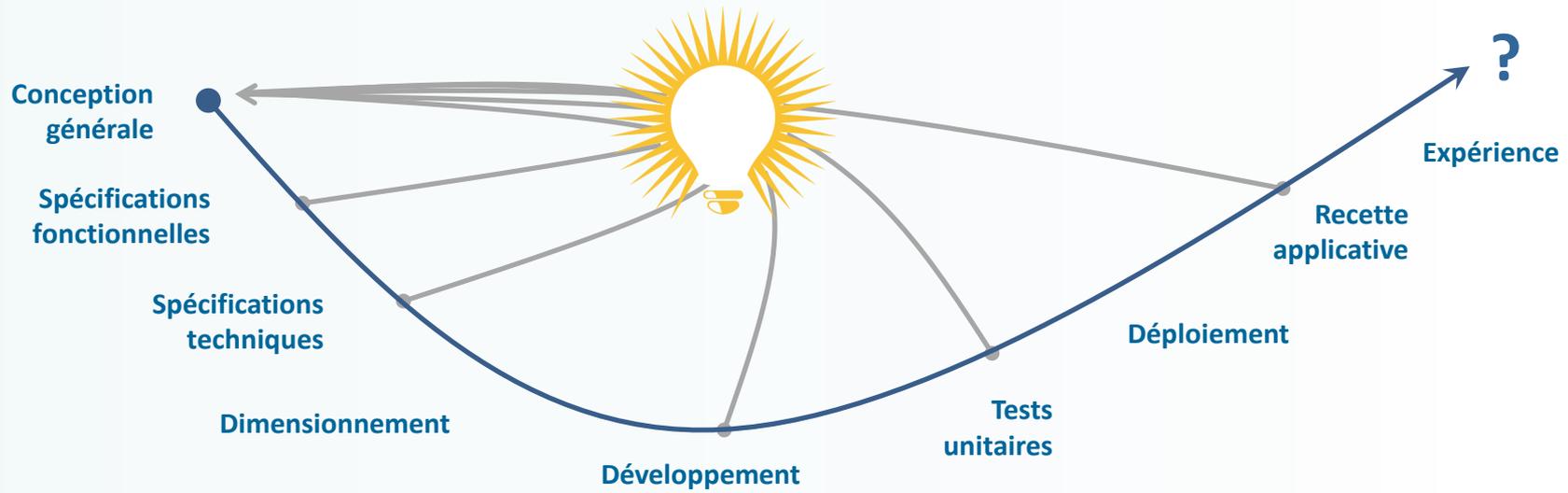
Le projet (contexte & objectif)

Un autre mode *projet*

REALISE



BESOIN

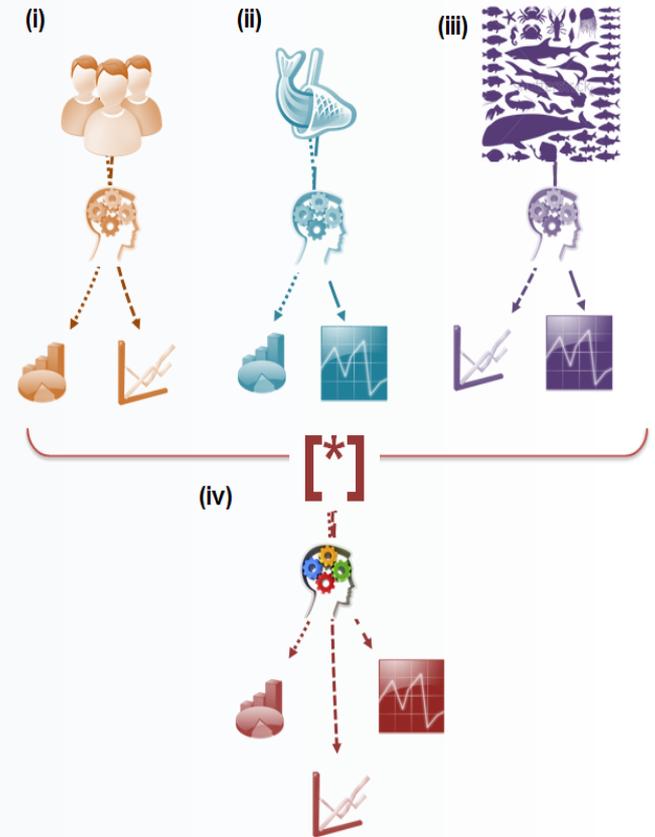




Produire une boîte à outils **interfacée et intégrée** permettant de **mutualiser** des **fonctionnalités de traitement de données et de jeux de données**.

Cet outil est **modulable et interopérable**.

Il permet la **construction progressive de modèles de classes**, sous forme de code informatique, pouvant être réadaptées d'un projet à un autre.





Ordinateur distant Utilisateur

Pas de logiciel
spécifique installé
(navigateur)

Peu d'éléments
exécutés côté client
pour garantir un
accès égal aux
traitements (hormis
restitution
cartographiées et
datavisualisation)

Accès via navigateur
(processus HTML)



Serveur distant dit *Plateforme de traitement de données*
Mis en place et maintenu par le projet *Données pour la décision*



Interface Web
Contient notamment
les formulaires de
configuration et
d'exécution des jobs



Configuration *utilisateur*
Métadonnées
Commentaires, posts

Configuration des jobs
dits *Scénarios*

Dépôt et récupération
des fichiers à traiter



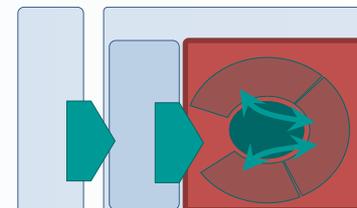
Base de données
Utilisateurs & frontspace

**Service de
Traitement
des
données**

Base de données
**Jeux de données
stockés**

Répertoire
**Fichiers
utilisateurs**





Technologie(s) générale(s)

JAVA (JSE7), PostGreSQL 9.4, PostGIS 2, XML / XSLT2

<xml />



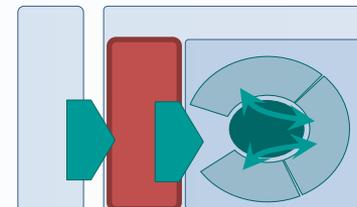
Principe d'utilisation

Réutiliser des standards existants : **le modèle applicatif est un *recomposeur*, une *glue* entre composants déjà développé par les différentes communautés.**

Ce modèle permet cependant l'injection d'algorithmes ou de correctifs de codes très spécialisés, « à la volée ».

Le langage utilisé par les wrappers est compréhensible par les métiers.





Technologie(s) générale(s)

Apache 2.4.9, PHP5, Javascript, PostGreSQL 9.4, XML / XSLT 2



Principe d'utilisation

L'interface est personnalisée, et permet l'édition de *billets* (c'est un gestionnaire de contenu avec quelques aspects sociaux).

L'utilisateur peut entretenir son **historique**. Les paramètres de **visibilité** et de **partage** sont assez complexes !

Beaucoup de choses sont **générées dynamiquement**. Les pages d'édition, les formulaires de lancement de scénarios, les réceptions de résultats par exemple.

Concernant l'aspect *javascript*, l'**exploitation des résultats** constitue le défi actuel. La plateforme se veut un **laboratoire** permettant aux ingénieurs de tester différentes solutions de représentations, issues de bibliothèques *opensource*.



Fonctionnalités, démonstration

Fonctionnalités cibles

Fonctions de connexion à des formats de fichiers

Fonctions de lecture "intelligente"	Alphanumérique, IG
Fonctions d'écriture de fichiers avec modèles	Alphanumérique, IG
Fonctions de captages de flux	Alphanumérique, IG, SGBD
Fonctions de téléchargement	Alphanumérique, IG

Fonctions de vérification

Intégrité de formats	Alphanumérique, IG, SGBD
Cohérence de séries de données	Alphanumérique, IG, SGBD

Fonctions de traitements et de transformations de données et de jeux de données

Transformation de fichiers à balises	Alphanumérique
Mobilisation de scripts	Alphanumérique, IG, SGBD
Gestion de fichiers	*
Conversion de format	*

Fonctions d'analyse de données

Recoupement de données	Alphanumérique, IG, SGBD
Attribution d'éléments spatialisés	Alphanumérique, IG, SGBD
Agrégation et jeux d'échelles	Alphanumérique, IG, SGBD

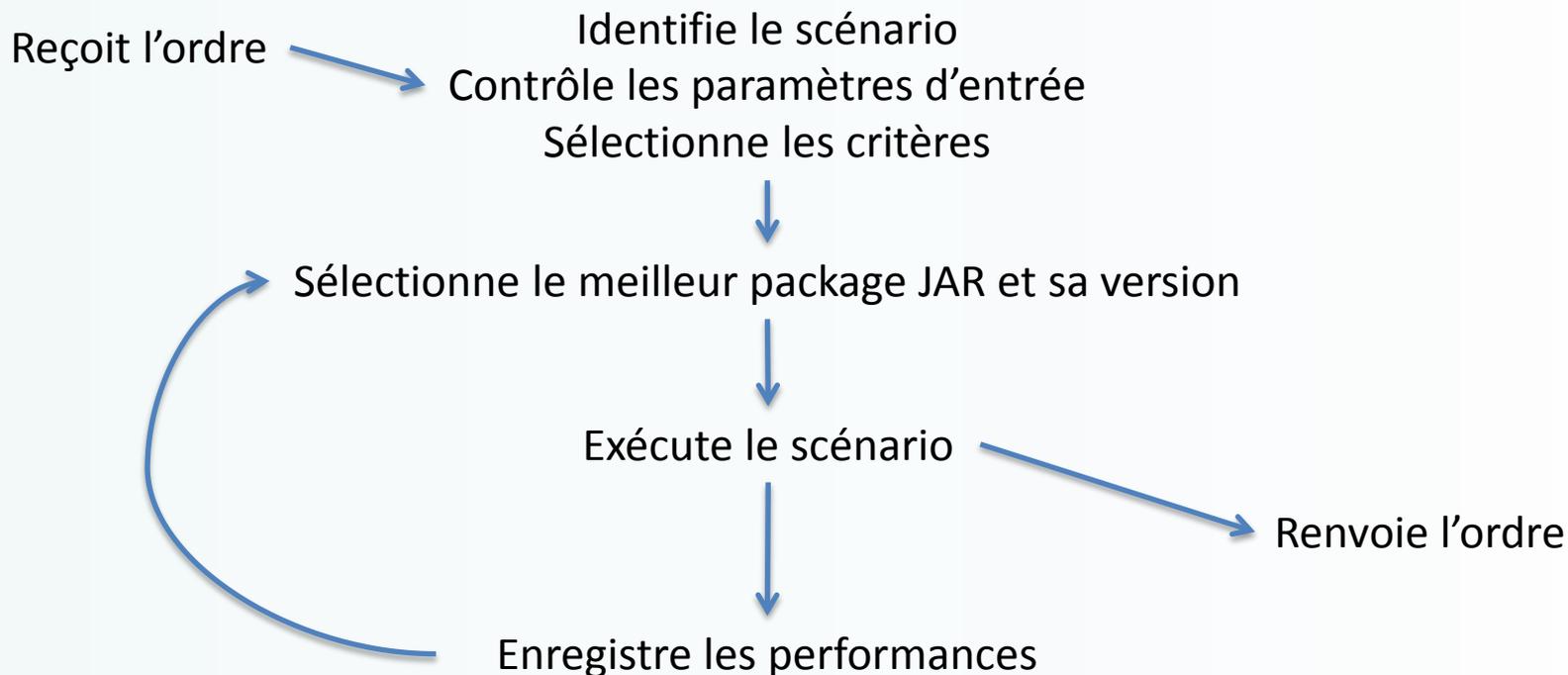
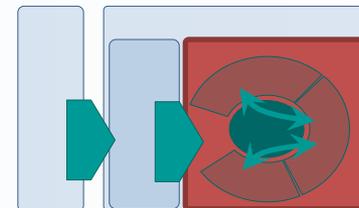


Des déploiements modulaires

Les déploiements de plateformes ne sont pas tous identiques.

Le service appelle des JAR contenant les modèles applicatifs, et ne comporte que les fonctionnalités offertes par les JAR embarqués.

Lorsqu'il existe plusieurs versions d'une même fonctionnalité, **un processus d'apprentissage** sera utilisé (en cours de développement) :





Free as in Freedom

Licence GPL3



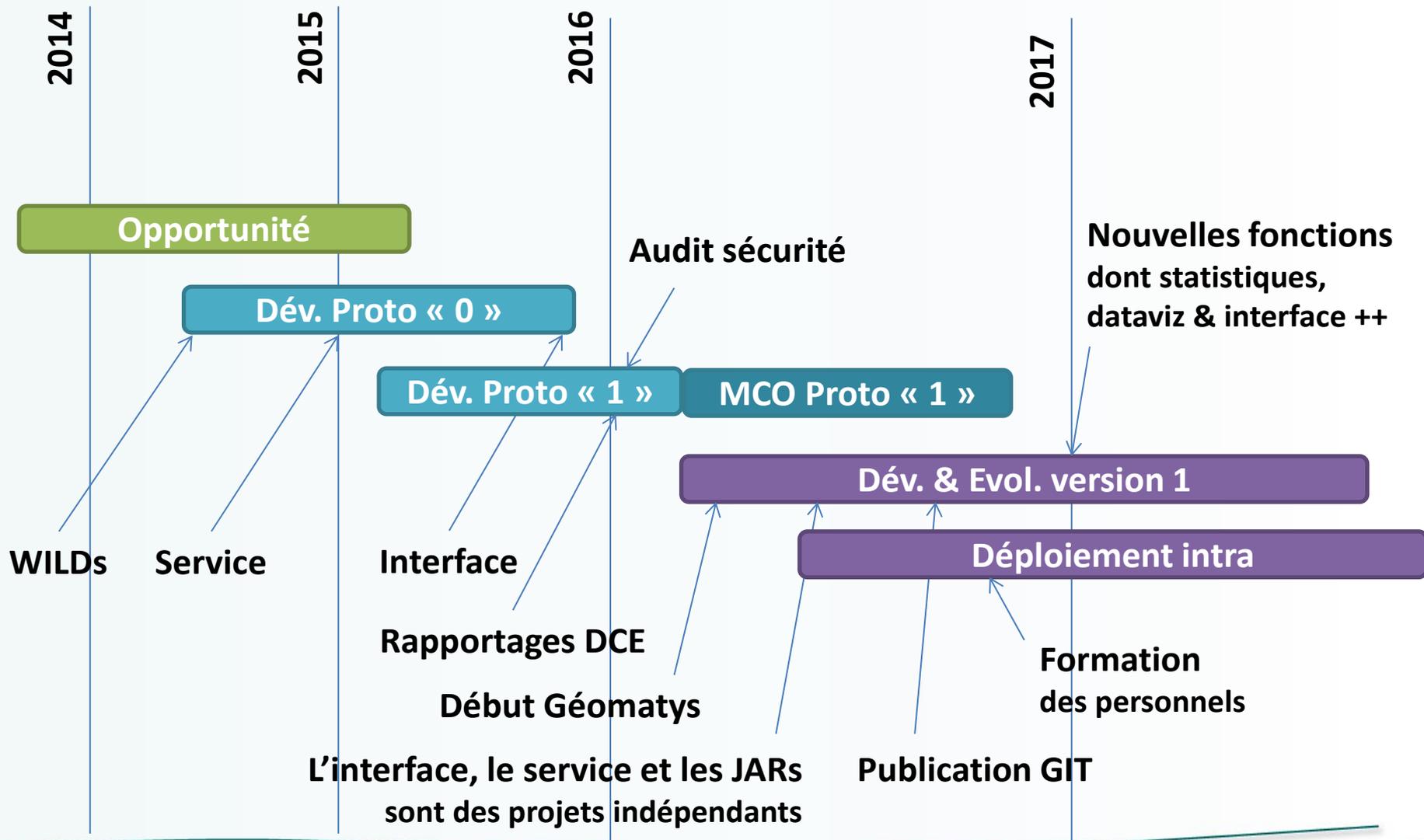
Ouverture d'un GIT prévu en S2 2016 – Nom de code **WILD**



Le projet initial est bien le développement d'une **boîte à outil ETL** qui permettra aux ingénieurs de mobiliser davantage de fonctionnalités OpenSource avec une ergonomie et une traçabilité accrues.



Perspectives RoadMap





L'équipe projet

Plateforme de traitement de données

- WILD -

Laurent Coudercy

Chef de département
Directeur du Pôle INSIDE

Alexandre Liccardi

Chef de projet

Jean-Philippe Goyen

Chargé d'études *Données du SIE*

Laurent Breton

Référent IG



Appui technique
Développements
depuis avril 2016

alexandre.liccardi@onema.fr

Merci de votre attention



Traiter les données : wrappers « REAL », le modèle 0

Modèle de développement JAVA utilisé.

- Classe développée par le projet *Données pour la décision*
- Classe développée par le prestataire, à partir du squelette fourni
- Classe complétée par le prestataire (ajout de méthodes)

REflexion
Apliance for the
Laziest

REAL est un package important :
il permet d'associer les éléments
JAVA au système de fichier et/ou
aux services web.

