

LOU TEEC E

26 OCTOBRE 2016
PIERRE LEVY
MAIRIE DE PARIS

PRÉSENTATION DE LA PLATE-FORME
ET DES EXEMPLES DE MISE EN ŒUVRE



AeGID Fondation



Municipalité de la Marsa



Université SESAME



Historique

- Démarrage du projet en juin 2001
- Choix d'architecture depuis l'origine
 - Back office orienté utilisateur métier et fonctionnel
 - Pas de couplage entre le contenu et la présentation
 - Architecture modulaire basée sur un noyau et des plugins
 - Utilisation exclusive de technologies open source et de formats interopérables
- Reversé dans le libre sous licence Open Source depuis 2002
- En 2008, utilisé par de portails à fort trafic : Météo France, AtosWorldline
- Base d'un progiciel éditeur depuis 2012

Qu'est-ce qu'un Logiciel Libre ?

- Lutece est un logiciel libre c'est à dire qu'il vous garantit les libertés suivante:
 - La liberté d'utiliser le logiciel, pour quelque usage que ce soit (liberté 0)
 - la liberté d'étudier le fonctionnement du programme, et de l'adapter à vos propres besoins (liberté 1). L'accès au code source est une condition pour tout ceci
 - La liberté de redistribuer des copies de façon à pouvoir aider votre voisin (liberté 2)
 - la liberté d'améliorer le programme, et de diffuser vos améliorations au public, de façon à ce que l'ensemble de la communauté en tire avantage (liberté 3). L'accès au code source est une condition pour tout ceci
- Lutece a été reversé dans le Libre en septembre 2002 sous une licence BSD. C'est le premier logiciel reversé par une collectivité ou une administration en France.

Qu'est-ce que Lutece ?

- Une plate-forme à la fois CMS, Portail et Framework
- Une architecture très modulaire noyau + plugins
 - + 400 plugins génériques ou métiers disponibles,
- Une expérience de plus de 15 ans
 - Expérience tant en matière de cas d'utilisations que d'exploitation
- Un socle technique basé sur des technologies reconnues
 - Socle JAVA utilisant les meilleurs frameworks Open Source (Spring, Freemarker, Bootstrap...)
- Une interface Front et Back Office adaptée à la consultation sur mobiles (Responsive Design)
- Des outils de déploiement permettant le Continuous Delivery

L'écosystème Lutece

Utilisateurs



Integrateurs



l'E-administration durable

Qui développe ?

Factoids and Stats



Lutece

- Mostly written in Java
- Mature, well-established codebase
- Large, active development team
- Well-commented source code
- Stable Y-O-Y development activity
- 25 active contributors

Commit Activity Timeline:



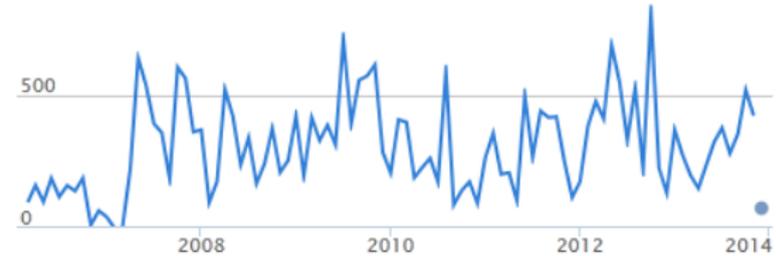
Updated Sep 15, 2014

more at [Open HUB](#)



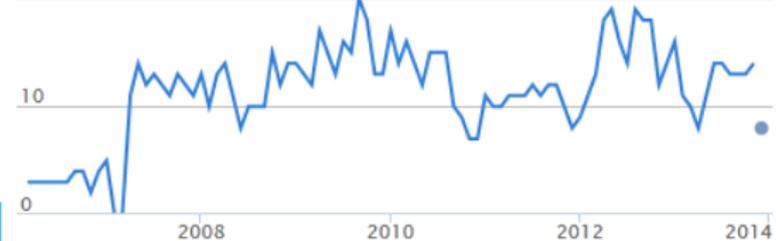
Commits per Month

1,000



Contributors per Month

20



Les intégrations technologiques

Search Engines



Authentication



Monitoring



ERP, ECM



Open data



CMIS



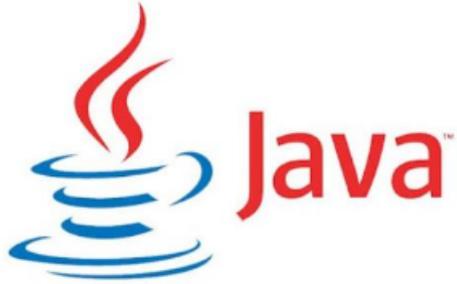
Machine learning



Quelle cible ?

DESIGNED TO BUILD LARGE AND COMPLEX PORTAL

WITH



maven



Jenkins

One command line to build and deploy a webapp using a large set of plugins for a given target (production, dev, integration, ...):

```
mvn clean lutece:site-assembly -P dev
```

Comment utiliser Lutece ?

- Trois principaux modes
 - "Out of the box" pour des modules standards
 - CMS, FAQ, Wiki, Digg, Petites annonces, ...
 - Assemblage de composants génériques
 - Plate-forme Formulaire/Worflow
 - Développements spécifiques
 - Pour les services numériques complexes et les applications métiers

Exemples « Out of the box »

- Sites Internet

- Site de LUTECE

<http://fr.lutece.paris.fr>

Exemples de fonctionnalités mises en œuvre : CMS, Forum, Wiki

- Site de démonstration

<http://dev.lutece.paris.fr/site-demo>

- Site de la médiation

<https://mediation.paris.fr/mediation/>

- Le Webapp Store

<http://fr.lutece.paris.fr/fr/app/appstore>

Assemblage de composants

- Téléservices usager avec guichet de suivi des demandes
 - Demandes d'urbanisme (Terrasses, enseignes, ...),
- Appel à projets
 - Appel à projets complexes avec instruction dans un back office : ex Paris Jeunes Talents.
- Formulaire de demandes avec workflow d'instruction
 - Autorisations de stationnement sur la voie publique pour déménagement, Permis de végétaliser, ...

<https://teleservices.paris.fr/k10/>

- Gestion des demandes des EPLE, Intranet logistique ...

Assemblage de composants

- Demandes de RDV avec workflow d'instruction

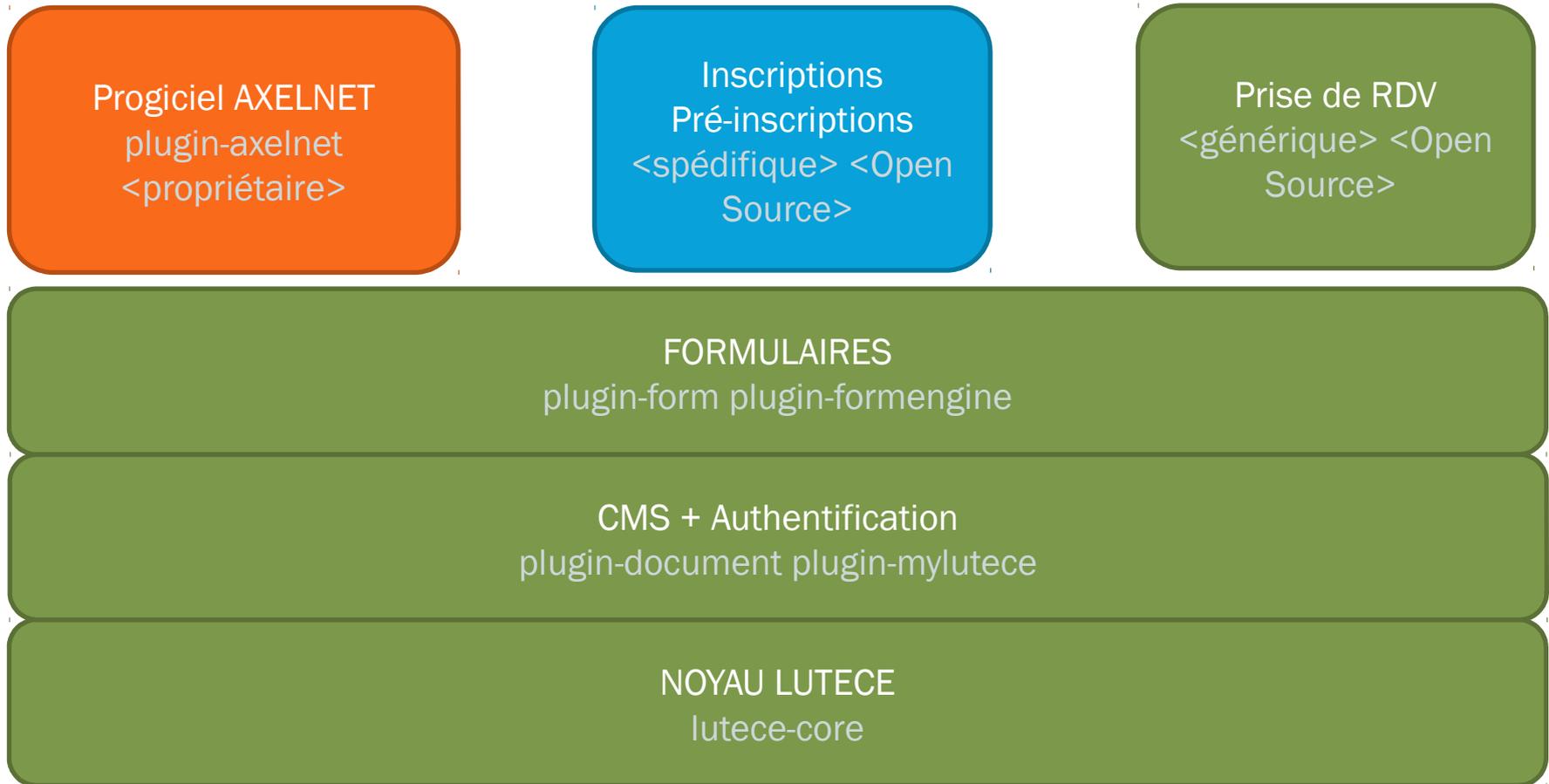
<https://teleservices.paris.fr/rdvssvp/>

<https://teleservices.paris.fr/rdvdu/>

<https://teleservices.paris.fr/rdvdases>

<https://teleservices.paris.fr/rdvma17>

Cas du Portail Familles



Développements spécifiques

■ Gestion administrative et usagers

- Gestion des délibérations, Gestion des documents du Conseil de Paris, Demandes d'actes d'état Civil, Gestion des aires de sport, Gestion des concours et examens professionnels, Budget Participatif, e-Pétitions, Billetterie Jeunes, Petites annonces

■ Gestion financière

- Gestion portail fournisseurs (SAP), Gestion des achats (SAP), Taxes de balayage, Taxes de séjour, Plan prévisionnel d'investissement, Archivage des régies.

■ Gestion de l'espace public

- DansMaRue (prix Territoria 2013), Gestion des cimetières, Gestion des arbres (prix Territoria 2014), Enlèvement des encombrants, Simulateurs de taxes enseignes et terrasses (prix Territoria 2016)

Exemples de réalisations

- Le Budget Participatif

<https://budgetparticipatif.paris.fr>

- Le simulateur de taxe « enseignes et terrasses »

<http://simutax-voirie.paris.fr>

- Logements sociaux du domaine municipal

<https://teleservices.paris.fr/locannonces/>

Exemples de réalisation

- La carte citoyenne

<https://teleservices.paris.fr/cartecitoyenne>

- Les e-Pétitions

<https://petition.paris.fr/epetition/>

- Le site de la médiation

<https://mediation.paris.fr/mediation/>

Le projet DansMaRue

- 2 applications mobiles (Android + iOs)
 - Disponibles sur les stores
- 1 formulaire web

<https://teleservices.paris.fr/sira/jsp/site/Portal.jsp?page=formengine&form=signalement>

- 1 API ouverte
- 1 back-office
 - Affectation automatique des signalements en fonction de leur nature et de leur position
 - Workflow d'instruction

Le projet DansMaRue

- 4 directions métiers : Voirie, Espaces verts, Propreté, Protection
- Près de 1000 utilisateurs du back-office
- Plus de 5000 signalements par mois
- Couplage avec la gestion des encombrants

Le projet DansMaRue

DansMaRue
plugin-dansmarue

Encombrants
plugin-ramen

WORKFLOW
plugin-workflow

GESTION UTILISATEUR / ENTITE ORGANISATIONNELLE
plugin-unittree

NOYAU LUTECE
lutece-core

La Gestion Relation Usager

- Au cœur du projet du Compte Parisien
- Choix du socle technologique a partir de PoC (Proof of Concept) réalisés sur 3 solutions Open Source
 - Publik de la société Entrouvert
 - CapDemat Evolution de l'association CapDemat
 - Lutece de la DSTI
- Deux grandes fonctionnalités mises en œuvre dans les PoC
 - La gestion des sollicitations : demandes d'informations / réclamations
 - La vue 360° d'un usager destinée aux agents

La Gestion Relation Usager

■ La problématique

- La gestion de la relation usager se gère au cœur des workflows back office des services numériques et des applications métiers
- Chacun des workflows disposent de sa propre population d'utilisateurs, associée a une gestion de droits et d'attributs spécifique. Ex : Dans Ma Rue
- Une solution de gestion centralisée des workflows n'est envisageable dans le contexte de la ville de paris tant sur le plan technique

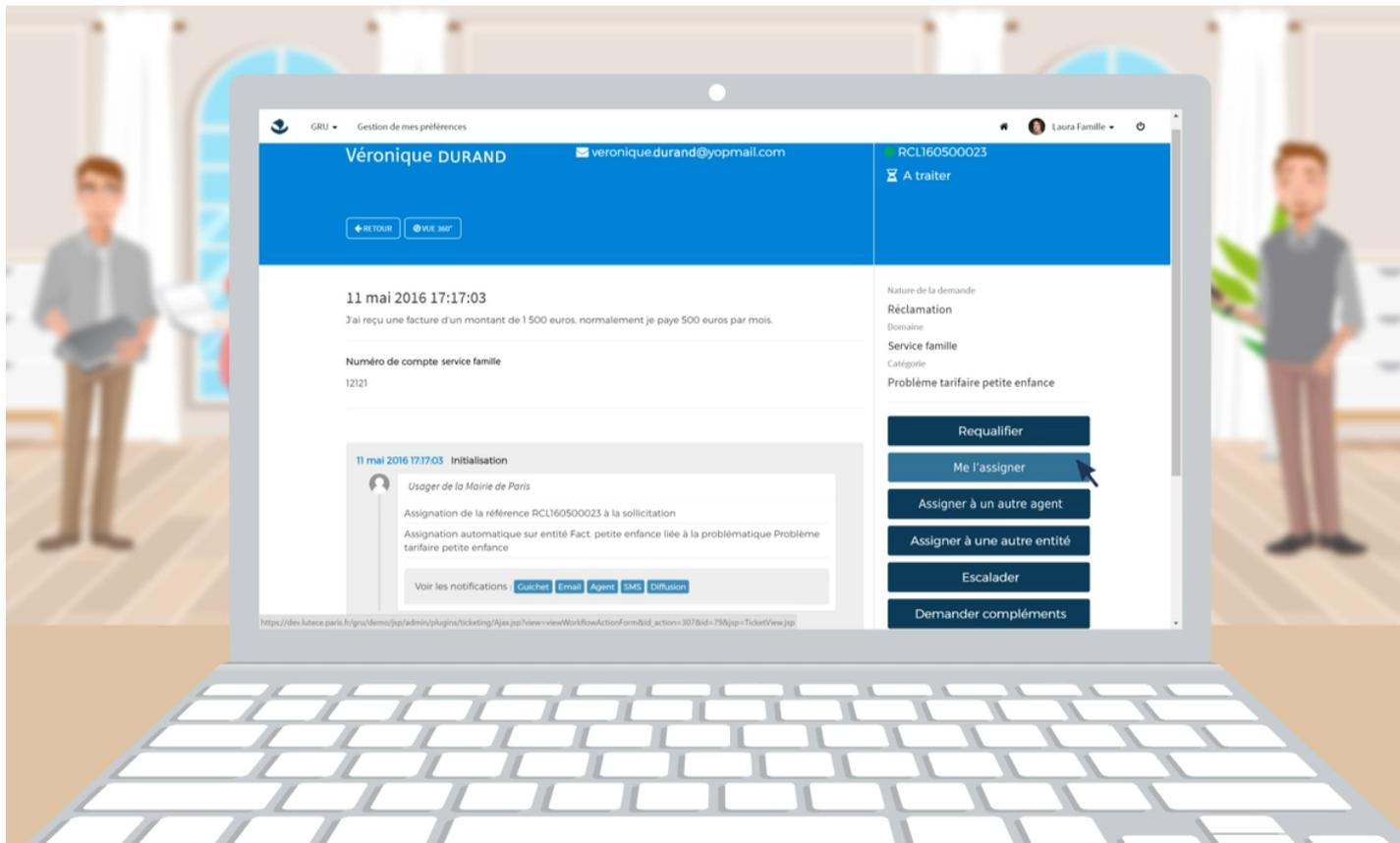
La Gestion Relation Usager

■ La solution

- La vision 360° de la Gestion de la Relation Usager doit s'appuyer sur les workflows des back offices (télé-procédures ou application métiers)
- Cette vue 360° doit être entièrement dé-corrélée des processus.
- Son alimentation doit se faire par un système de collecte normalisé : typiquement une API ouverte à toutes les technologies.
- La vue 360° doit permettre d'intégrer facilement des liens vers la création de nouvelles demandes ou vers les back office des demandes en cours en profitant si possible de mécanismes de Single-Sign-On.

La Gestion Relation Usager

Vidéo



FranceConnect – Etat Plateforme



- Les principes
- PoC réalisé lors du Hackathon

<https://dev.lutece.paris.fr/hackathon-dev/>

- Les exemples de code

<https://franceconnect.gouv.fr/code-fs>

Les atouts de Lutece

- Réponse urbanisée au foisonnement des demandes en matière de portails et de services
 - Usine à sites basée sur des référentiels de gestion de configuration,
 - Portail d'intégration éditorial/applicatif, Séparation forme/contenu, normalisation des chartes graphiques
- Relation à l'usager et à son/ses identité(s)
 - Ensemble de plugins guichet/formulaires/workflow/vue 360 interfacés avec les principaux systèmes d'authentification et de GID
- Respect des lois s'appliquant aux sites publics
 - Conformité RGAA, RGI, RGS et recommandations CNIL

Les atouts de Lutece

- Stratégie Open Source et respect des standards ouverts
 - Indépendance vis à vis des éditeurs, ouverture aux intégrateurs
 - Solution reconnue par de grands intégrateurs :
 - Sopra, Atos, ACN, Consortium OW2 (Quality Award 2016)
- S'adapter aux nouveaux usages
 - Mobilité : Full Responsive Design
- Garantir la meilleure utilisation des deniers publics
 - Pas de frais de licence, socle de mutualisation technique et fonctionnelle. Exemple : Marseille !

Les atouts de Lutece

- Interface utilisateur et ergonomie standard
 - Gains en spécification et en formation
- Sécurisation du code
 - Mêmes règles de développement et contrôle Qualité
 - Systèmes de gestion de versions SVN et GIT
- Simplification des déploiements
 - Mêmes procédures d'installation et d'exploitation
 - Possibilité d'automatisation : DevOps / NoOps
 - => Réconciliation DEV <-> PROD

Quid de la gouvernance ?

- La modularité de Lutece est LA réponse
 - Chacun peut adapter le produit avec un minimum d'impact
 - Il est tout à fait possible de « forker » un plugin, un module de plugin ou une implémentation d'une interface.
- Cette modularité est apportée par :
 - La conception de l'architecture de plugins de Lutece basée sur les possibilités du langage (Java) :
 - Interface, API, Polymorphisme
 - Instanciation dynamique, introspection
 - Injection de dépendances (Spring)
 - Les techniques de build (Maven) qui garantissent un assemblage intelligent de tous les composants.

Le bon choix Open Source

- Tout logiciel Open Source n'est pas nécessairement un bon choix !
- Les critères de maturité et les fonctionnalités
 - Taille et activité de la communauté des développeurs
 - Support de fonctionnalités avancées : accessibilité, internationalisation, ...
 - Adaptabilité : Responsive design
 - Sécurité : DOS , Cross-Scripting, injection SQL
 - Interopérabilité (capacité à communiquer et à s'intégrer avec d'autres produits)
 - Documentation (utilisateur, intégrateur, développeur, exploitant)
 - Maintenabilité (capacité à faire intervenir plusieurs intégrateurs)
 - Exploitabilité (installation, monitoring, logs et debug)
 - Performances (caches, optimisations)

Comment contribuer ?

■ Niveau 0

- Utiliser l'outil et le faire savoir

■ Niveau 1

- Remonter des anomalies (JIRA), proposer des évolutions (JIRA), contribuer aux documentations (Wiki)

■ Niveau 2

- Fournir des corrections et des évolutions
 - Patch sous forme de fichier attaché dans un JIRA
 - Pull Request sur GitHub
- Modification directe du code si les personnes sont habilitées

■ Niveau 3

- Créer une équipe et des projets dans le repository GitHub lutece-secteur-public

Les référentiels de code Open Source

- Code source de tous les composants ouvert sur GitHub :

<https://github.com/lutece-platform>

- 139 référentiels 46 contributeurs : Noyau, CMS, Moteur de Formulaire, WIKI, FAQ, les modules d'authentification, SEO, mesure d'audience G. Analytics, Piwik, ...

<https://github.com/lutece-secteur-public>

- 64 référentiels 32 contributeurs : Gestion de la Relation Usager, Demandes de RDV, Consultation du Plan Local d'Urbanisme, Contrôle de Légalité, ...
- Liste des composants :

<http://fr.lutece.paris.fr/fr/jsp/site/Portal.jsp?page=components>

- Toute entité (intégrateur/collectivité) peut créer sa propre équipe de contributeurs et déposer ses propres composants sur LuteceSecteurPublic.

MERCI !

- Nous suivre sur Twitter : [@LuteceNews](https://twitter.com/LuteceNews)

