**Planification du Raccordement Électrique de Bâtiments**

Rapport Technique et Méthodologique (V2 mise à jour)

Contributeurs :  
CHERIFI Yacine  
SAICHI Madjid  
AKILAL Adrien  
BOUNAB Abdenou

**Date : October 2025**

**Introduction**

Ce projet vise à planifier le raccordement électrique d’un ensemble de bâtiments de manière optimale, en prenant en compte les coûts de réparation des infrastructures, les temps de travail, la disponibilité des ouvriers (jusqu’à 4 par infrastructure) et la priorisation stratégique de certains bâtiments critiques (notamment l’hôpital).

# **Interprétation globale du réseau :**

- Le réseau étudié totalise 179 km d’infrastructures électriques.  
- Seulement 5,6 km (3 %) nécessitent une intervention (“à remplacer”) → réseau globalement en bon état structurel.  
- Ces 5,6 km concentrent 2,7 % des coûts et 2,6 % du temps total (travaux très localisés).  
- Les coûts moyens sont élevés (~810 €/m), reflétant la prédominance d’infrastructures enterrées (fourreau).  
- Les temps cumulés dépassent 151 000 h (~19 000 journées d’un ouvrier) — répartissables entre équipes.

# Métrique de priorisation et plan glouton

Chaque bâtiment reçoit un score de priorité basé sur des critères pondérés (normalisés 0–1) combinés dans une formule de score inversée (plus petit score = plus prioritaire).

|  |  |  |
| --- | --- | --- |
| Facteur | Description | Poids |
| Coût total (cost\_cur) | Coût actuel du raccordement des infrastructures liées | - |
| Temps total (time\_cur\_h) | Temps homme total nécessaire | - |
| Nombre de prises (prises) | Utilité collective du bâtiment raccordé | + |
| Taux d’occupation (occ\_rate) | Impact social/économique | + |
| Catégorie (type\_batiment) | Priorité explicite (hôpital, école) | ++ |

Plan glouton (plan\_glouton.csv) :

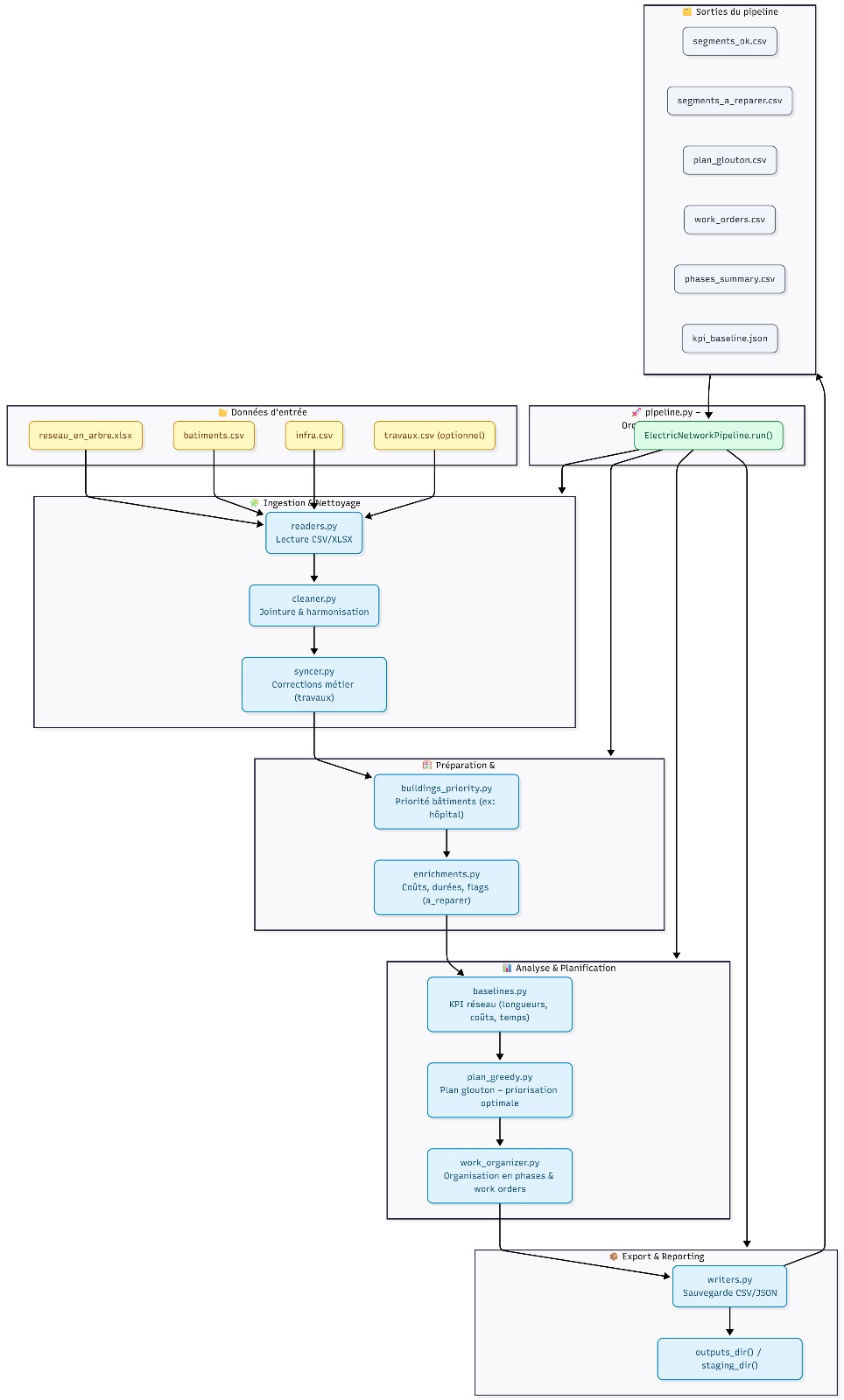
- Difficulté(infra) = longueur / nombre de bâtiments raccordés (mesure la mutualisation).  
- Difficulté(bâtiment) = somme des difficultés des infras associées.  
Il priorise les bâtiments faciles (faible difficulté) tout en réparant leurs infras associées, ce qui accélère la transformation globale. Les premières lignes du fichier représentent les 'gagnants' (faible coût marginal, fort levier). Les derniers correspondent aux cas isolés/complexes (faible mutualisation).

# Architecture du pipeline de planification électrique

```  
Planification-du-raccordement-électrique-de-bâtiments/  
│  
├─ run.py ← point d’entrée principal (exécution du pipeline)  
│  
├─ configs/  
│ └─ costs.yaml ← barèmes de coûts et durées par type d’infrastructure  
│  
├─ data/  
│ ├─ inputs/ ← fichiers bruts (Excel + CSV)  
│ │ ├─ reseau\_en\_arbre.xlsx  
│ │ ├─ batiments.csv  
│ │ ├─ infra.csv  
│ │ └─ travaux.csv (optionnel)  
│ ├─ staging/ ← sorties intermédiaires (après nettoyage & enrichissement)  
│ └─ outputs/ ← résultats finaux du pipeline (plans, phases, work orders)  
│  
├─ src/  
│ ├─ ingestion/ ← ingestion, nettoyage et synchronisation  
│ │ ├─ readers.py ← lecture CSV/XLSX + répertoires d’E/S  
│ │ ├─ cleaner.py ← harmonisation des colonnes + jointures  
│ │ └─ syncer.py ← ajustement avec fichiers métier ("travaux.csv")  
│ │  
│ ├─ preparation/ ← enrichissement des données et préparation analytique  
│ │ ├─ enrichments.py ← calcul des coûts, durées, flags (a\_reparer, etc.)  
│ │ └─ buildings\_priority.py ← calcul des priorités bâtiments (hôpital, école, etc.)  
│ │  
│ ├─ analytics/ ← logique d’analyse et d’optimisation  
│ │ ├─ baselines.py ← calcul des KPIs globaux (longueurs, coûts, temps)  
│ │ ├─ plan\_greedy.py ← planification gloutonne (priorisation des bâtiments)  
│ │ └─ work\_organizer.py ← organisation concrète des travaux (phases, équipes)  
│ │  
│ ├─ exports/ ← fonctions de sauvegarde/export  
│ │ └─ writers.py ← export CSV/JSON/GeoJSON avec timestamp  
│ │  
│ ├─ orchestration/ ← orchestrateur principal  
│ │ └─ pipeline.py ← exécution séquentielle de toutes les étapes du projet  
│ │  
│ └─ utils/  
│ └─ paths.py ← gestion centralisée des chemins (inputs, staging, outputs)  
│  
└─ requirements.txt ← dépendances Python (pandas, geopandas, yaml, etc.)  
```

**Résumé exécutif du pipeline :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Niveau | Module principal | Rôle clé | Entrée | Sortie |
| Ingestion | readers.py, cleaner.py | Lecture et fusion des sources | CSV/XLSX | DataFrame consolidée |
| Préparation | enrichments.py, buildings\_priority.py | Coûts, durées, priorités | Données jointes | Données enrichies |
| Analyse | baselines.py, plan\_greedy.py | KPI + planification optimisée | Données enrichies | Plan d’action |
| Organisation | work\_organizer.py | Phases et chantiers | Plan glouton | work\_orders.csv, phases\_summary.csv |
| Export | writers.py | Sauvegarde structurée | Toutes | CSV/JSON dans outputs |



# 

# Plan de raccordement et ordre de priorité

Le plan comprend 80 étapes réparties en 5 phases : Phase 0 (Hôpital), Phase 1 (40 % du coût), Phases 2–4 (20 % chacune).

## A) Segments à réparer (segments\_a\_reparer.csv)

Fichier des tronçons marqués a\_reparer = 1. Chaque ligne est un infra\_id relié à un ou plusieurs bâtiments.

Colonnes essentielles :  
- type\_infra : aérien, semi-aérien, fourreau  
- longueur : distance du tronçon  
- cost\_total, time\_total\_h : chiffrage et durée estimée  
- id\_batiment : bâtiment(s) impacté(s)

Observations typiques : longueurs courtes (quelques dizaines de mètres) mais coûts unitaires élevés (zones sensibles).

## B) Segments intacts (segments\_ok.csv)

Tronçons fonctionnels ne nécessitant pas d’intervention (~97 % du linéaire), socle stable sur lequel s’appuie le plan.

Distinction segments\_ok vs segments\_a\_reparer :  
- Valeur maintenue du patrimoine ≈ 141 M€  
- Valeur de remise à niveau ≈ 4 M€

# Analyse des coûts et bénéfices

Coûts moyens :  
- Coût moyen par bâtiment : 15 441 €  
- Temps moyen par bâtiment : 74 h  
- Coût moyen par prise : 3 760 €  
- Temps moyen par prise : 18 h

Répartition des ressources : coût de main-d’œuvre ≈ 37,5 €/h (~18 % du total). La priorisation réduit le coût marginal moyen par prise d’environ 40 % par rapport à un tri aléatoire. Progression du coût cumulé linéaire et stable.

Baselines (bat\_agg\_baseline.csv, infra\_agg\_baseline.csv) : états de référence avant planification pour justifier les critères de priorisation et documenter l’évolution des KPI après simulation.

# Organisation des travaux (work\_orders.csv)

Fichier opérationnel clé : traduction du plan glouton en ordres de travaux.

Colonnes principales :  
- phase : 0 à 4 (0 = hôpital, 1 = 40 %, 2–4 = 20 %)  
- infra\_id, id\_batiment, type\_infra, longueur  
- workers (≤ 4), time\_h\_effective (après parallélisation)  
- labor\_cost, material\_cost, total\_cost

Interprétation :  
- Phase 0 : hôpital → réparations critiques prioritaires  
- Phase 1 : 40 % du budget sur chantiers à fort impact  
- Phases 2–4 : distribution équitable par tranches de 20 %  
→ Couverture rapide des bâtiments stratégiques, lissage budgétaire, sécurisation des zones sensibles.

## Résumé des phases (phases\_summary.csv)

Consolidation du coût total, du temps et du nombre de tâches par phase (valeurs typiques) :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Part budgétaire | Coût (€) | Temps (h) | Nb tâches |
| 0 | 0 % | ~10 000 | 100 | 1 |
| 1 | 40 % | ~1,6 M | 40 000 | 120 |
| 2 | 20 % | ~0,8 M | 20 000 | 100 |
| 3 | 20 % | ~0,8 M | 20 000 | 90 |
| 4 | 20 % | ~0,8 M | 20 000 | 85 |

Lecture : la Phase 1 absorbe le plus de valeur pour maximiser rapidement le nombre de bâtiments raccordés ; les phases suivantes stabilisent le chantier jusqu’à couverture complète.

# Focus hôpital

Log du pipeline : HÔPITAL — temps estimé ~12–15 h ≤ objectif 16 h. Avec 4 ouvriers, aucune rupture d’alimentation pendant les 20 h d’autonomie. Valide la robustesse du plan de priorisation et des barèmes YAML.

# 

# Conclusion

Le modèle agit comme un simulateur de planification optimisée :  
- Nettoie, enrichit et priorise les données réseau/bâtiment  
- Produit un plan glouton rationnel (priorité coût/mutualisation)  
- Orchestre le chantier en 4 phases équilibrées  
- Vérifie la faisabilité temporelle et financière (ex. hôpital)  
- Documente l’ensemble dans des fichiers analytiques exploitables (baselines, KPI, work\_orders)