

Immobilisations et amortissement

Responsabilité

Les immobilisations couvrent la génération des achats d'immobilisations, leur durée de vie, leur mode d'amortissement, les rapports d'immobilisations, les taxes associées et les écritures de régularisation.

Static data

La configuration principale se trouve dans `immobilizations.json` et les settings/modules qui activent les préférences liées aux immobilisations. La page **Configuration Accounting — guide des JSON** décrit la source de configuration.

Serveur

- `classes/bills/immobilizations/ImmobilizationsBill.js` : bill d'acquisition;
- `ImmobilizationsTaxesBill.js` : taxes municipales/scolaires ou autre taxe configurée;
- `reports/ImmobilizationsReportBill.js` : rapport basé sur les immobilisations référencées;
- `utilsByModule.getImmobilizationDepreciation()` : calcul d'amortissement selon le type d'unité;
- `BillCreationContext._usedImmobilizationsByType` : contexte utilisé pendant la génération.

Modèles Unity

- `ImmobilizationData`;
- `SeededImmobilization` et `SeededImmobilizationPlus`;
- `BillDataImmobilization`;

- `ImmobilizationBillEntry`;
- `ImmobilizationTaxe` et `ImmobilizationTaxesBillEntry`.

UI

Les rapports possèdent des contrôles dédiés : `ImmobilizationsReportControl`, variante Plus, rapports dégressive/by usage et `ImmobilizationsTaxesReportControl`. Le contrôle détecte notamment le type d'amortissement pour adapter le contenu affiché.

```
flowchart LR
    CFG[immobilizations.json] --> CREATE[Create Module Type Data]
    CREATE --> SEED[SeededImmobilization]
    SEED --> BILL[ImmobilizationsBill]
    BILL --> ACC[Accounts / Journal]
    SEED --> DEP[Depreciation calculation]
    DEP --> REPORT[Immobilizations Report]
    REPORT --> REG[Regularization entries]
```

Points d'attention

- Le type d'immobilisation détermine le compte impacté.
- Le calcul dépend des dates fiscales de l'activité.
- Les contextes suivants peuvent réutiliser les immobilisations précédentes.
- Un changement doit être testé dans création des données, answer key et grading.

Ajouter un type d'immobilisation ou une méthode d'amortissement

Nouveau type d'immobilisation

1. Ajouter/configurer l'entrée dans `immobilizations.json`.
2. Ajouter les comptes nécessaires dans le plan comptable si le type crée de nouveaux UUIDs : actif, depreciation, cumulative depreciation, etc.
3. Vérifier la génération dans Create Module Type Data et `ImmobilizationsBill`.

4. Vérifier `ImmobilizationsReportBill` et les données Unity `SeededImmobilization*`.
5. Vérifier le report UI correspondant.
6. Vérifier les relations d'Answer Key et Grading qui construisent les écritures d'amortissement.
7. Tester le passage au contexte suivant et la dépréciation cumulative.

Nouvelle méthode d'amortissement

Le calcul central est `utilsByModule.getImmobilizationDepreciation()`. Les branches existantes sont `by_usage_km`, `linear` et `degressive`.

1. Ajouter le nouvel `unit_type` dans le calcul central.
2. Définir précisément l'unité stockée dans `total_units_used`.
3. Mettre à jour la propagation entre contextes dans Create Module Type Data.
4. Adapter le report Unity s'il affiche des colonnes différentes.
5. Vérifier Answer Key et Grading, qui appellent tous deux `getImmobilizationDepreciation()`.
6. Tester acquisition avant, pendant et près de la fin de l'exercice.

Dates fiscales

Le calcul utilise `actSettings.year`, `startFiscalDateObj` et `endFiscalDateObj`. Un bug “un mois de trop/de moins” doit être reproduit avec les timestamps exacts et les bornes fiscales, pas uniquement avec le prix de l'actif.

Debug

Comparer dans l'ordre : seeded immobilization → bill entry → résultat de `getImmobilizationDepreciation()` → report → account relation → Answer Key → Grading. Cela localise rapidement si l'erreur vient de la génération ou de la correction.

Revision #2

Created 2026-09-18 19:17:33 UTC by Mathis

Updated 2026-09-18 20:13:37 UTC by Mathis