

# MiCetF Data - Package de données partagées

---

Ce package centralise toutes les données descriptives des applications du domaine MiCetF. Il permet une gestion unifiée des informations qui peuvent être partagées entre toutes les applications.

## Installation

### Dans le monorepo

Si vous travaillez dans le monorepo, les dépendances sont déjà gérées par les workspaces npm.

### Dans un projet externe

Si vous souhaitez utiliser ces données dans un projet externe, vous pouvez les installer via npm :

```
# Si vous avez publié le package sur npm :
npm install micetf-data

# Si vous utilisez un lien local :
npm install file:../path/to/packages/data
```

## API

### Applications

```
import {
  applications,
  getApplicationById,
  getApplicationsByDomain,
} from "micetf-data";

// Récupérer toutes les applications
console.log(applications);

// Récupérer une application par son ID
const fractions = getApplicationById("fractions");
console.log(fractions);

// Récupérer les applications d'un domaine spécifique
const mathsApps = getApplicationsByDomain("maths");
console.log(mathsApps);
```

### Domaines

```
import { domaines, getDomaineById, getAllDomaines } from "micetf-data";

// Récupérer tous les domaines
console.log(domaines);
// Alternative
console.log(getAllDomaines());
```

```
// Récupérer un domaine par son ID
const maths = getDomaineById("maths");
console.log(maths);
```

## Miniatures et images

```
import { getThumbnailUrl, thumbnailExists } from "micetf-data";

// Obtenir l'URL d'une miniature
const thumbnailUrl = getThumbnailUrl("fractions.png");
console.log(thumbnailUrl);

// Vérifier si une miniature existe
const exists = thumbnailExists("fractions.png");
console.log(exists);
```

## Utilitaires

```
import { filterToolsBySearchTerm, formatDate } from "micetf-data";

// Filtrer les outils par terme de recherche
const filteredTools = filterToolsBySearchTerm(applications, "fraction");
console.log(filteredTools);

// Formater une date
console.log(formatDate("2024-03-31")); // Affiche "31/03/2024"
```

## Constantes

```
import {
  VERSION,
  LAST_UPDATE,
  PAYPAL_ID,
  CONTACT_EMAIL,
  BASE_URL,
} from "micetf-data";

console.log(`Version des données: ${VERSION}`);
console.log(`Dernière mise à jour: ${LAST_UPDATE}`);
console.log(`ID PayPal: ${PAYPAL_ID}`);
console.log(`Email de contact: ${CONTACT_EMAIL}`);
console.log(`URL de base: ${BASE_URL}`);
```

## Compatibilité avec le code existant

Ce package maintient certaines fonctions pour la compatibilité avec le code existant :

```
import { outils, amis } from "micetf-data";
import { getFullThumbnailPath, getThumbnailPath } from "micetf-data";

// Récupérer les outils d'un domaine (alias de getApplicationsByDomain)
const mathsOutils = outils("maths");
console.log(mathsOutils);

// Récupérer les sites amis
console.log(amis);

// Utiliser les anciennes fonctions pour les miniatures
// (ces fonctions sont des alias de getThumbnailUrl)
const path1 = getFullThumbnailPath("fractions.png");
const path2 = getThumbnailPath("fractions.png");
```

## Structure des données

### Application

```
{
  id: "appli1",           // Identifiant unique
  title: "Application 1", // Titre
  description: "Description de l'application 1", // Description
  url: "/appli1/",       // URL relative ou absolue
  thumbnail: "appli1.jpg", // Nom du fichier miniature
  keywords: ["maths", "geometrie", "cm1", "cm2"], // Mots-clés/domaines
}
```

### Domaine

```
{
  name: "maths",           // Identifiant unique
  label: "Mathématiques",  // Libellé à afficher
  href: "#maths",          // Lien pour la navigation
  icon: "calculator"       // Nom de l'icône associée
}
```

## Intégration avec Vite

Ce package utilise `import.meta.glob` de Vite pour gérer les miniatures. Si vous n'utilisez pas Vite, vous devrez peut-être adapter la gestion des images.

## Mise à jour des données

Pour ajouter ou mettre à jour des applications ou des domaines, modifiez les fichiers correspondants dans le dossier `src/`. Ensuite, mettez à jour la version dans `src/constants.js`.