

# Procédure de création d'un graphique d'analyse data

## 0) But du document

Formaliser une procédure **reproductible** pour :

- importer et préparer les données (Power Query),
  - construire un **modèle relationnel fiable** (clés + relations + rôles),
  - créer les **mesures DAX** utilisées,
  - construire les éléments clés du dashboard (slicers, tooltips, visuels),
  - exécuter la **mise à jour hebdomadaire**.
- 

## 1) Contexte métier (règles de pilotage)

Sanitoral suit un portefeuille de projets sur 4 régions et 2 typologies (IT / Marketing), avec 3 KPI :

- **coûts**,
- **respect des deadlines / durée**,
- **livrables vs planning**.

### Règle d'alerte

Une **alerte** est déclenchée dès qu'il y a **15% d'écart** entre **prévisionnel** et **actuel** sur au moins un KPI.

### Utilisateurs cibles (3 niveaux)

- Directeur général (vision globale + arbitrage)
- Directeurs régionaux (vision filtrée sur leur région)
- Directeurs pays (vision filtrée sur leur pays, analyse projet/phase)

## 2) Contraintes projet

- Mise à jour **toutes les semaines** → nettoyage **automatique** via **Power Query** (“Étapes appliquées”).
  - Aucune modification manuelle du fichier source : tout se fait dans Power BI (Power Query + modèle).
  - (Option) utilisation d'un **Gantt** et/ou Q&A ; usage d'**info-bulles** recommandé.
- 

## 3) Données — tables présentes dans le modèle

### Tables “cœur”

- **Projects\_Plans** : table principale au niveau *projet/phase* (prévu, actuel, écarts, dates, etc.)
- **Alert** : statut d'alerte par phase (Cost\_Alert / Duration\_Alert / Deliverable\_Alert + écarts)
- **Project\_Type** : dimension IT / Marketing
- **Country\_Profiles** : dimension pays + attributs + rattachement à Region\_ID
- **Region** : dimension région

### Tables “sécurité / rôles”

- **User\_country\_profile** : (Country\_ID, LoginUser)
  - **User\_region\_profile** : (Region\_ID, LoginUser).
- 

## 4) Power Query — procédure de préparation (ETL)

Objectif : garantir une mise à jour hebdo **sans intervention manuelle**.

### 4.1 Import

1. Accueil → **Obtenir des données**
2. Sélectionner la source (Excel / CSV / SQL...)
3. **Transformer les données** (ouvrir Power Query)

### 4.2 Contrôles minimum (à appliquer aux tables utilisées)

À faire pour chaque table :

- **Types de colonnes** : Date / Nombre / Texte (indispensable pour DAX et relations)  
Oral P7 (1)
- **Valeurs manquantes / erreurs** : repérer les anomalies (qualité de colonne)

- **Nettoyage :**
  - standardiser noms de colonnes,
  - supprimer lignes/colonnes vides,
  - corriger anomalies évidentes,
  - vérifier cohérence des IDs.

### 4.3 Point clé : clé unique au niveau phase

Créer la clé **Propha\_ID** en concaténant **Project\_ID + Phase** (colonne calculée).

---

## 5) Modèle de données — relations et “pourquoi il tient”

Le risque majeur : **Project\_ID n'est pas unique** car un projet contient plusieurs phases.  
→ Solution : **Propha\_ID** garantit l'unicité au niveau projet/phase.

### 5.1 Relations principales

- **Alert ↔ Projects\_Plans via Propha\_ID**
  - Cardinalité : **1-1** (une alerte par phase)
  - Filtre : **double sens** (sélection phase ↔ alerte)  
Oral P7 (1)
- **Projects\_Plans → Project\_Type via Project\_ID**
  - Cardinalité : **N-1**
  - Filtre : **sens unique** (logique “je filtre un sous-ensemble de projets puis je lis la typologie”)  
Oral P7 (1)
- **Projects\_Plans → Country\_Profiles via Project\_ID**
  - Cardinalité : **N-1**
  - Filtre : **sens unique** (filtrer projets/phases → obtenir pays concernés)  
Oral P7 (1)
- **Country\_Profiles ↔ Region via Region\_ID**
  - Logique : plusieurs pays par région (relation plusieurs-à-un)

### 5.2 Relations “rôles” (RLS / filtrage par utilisateur)

- **Country\_Profiles ↔ User\_country\_profile via Country\_ID** (contrôle périmètre pays)
- **Region ↔ User\_region\_profile via Region\_ID** (contrôle périmètre région)

## 6) Mesures DAX — dictionnaire (tes noms exacts)

### 6.1 Totaux (base)

- **Total Planned Cost**  
`SUM(Projects_Plans[Planned_cost])`
- **Total Actual Cost**  
`SUM(Projects_Plans[Actual_cost])`
- **Total Planned Duration**  
`SUM(Projects_Plans[Planned_duration])`
- **Total Actual Duration**  
`SUM(Projects_Plans[Actual_duration])`
- **Total Planned Deliverable**  
`SUM(Projects_Plans[Planned_delivrable])`
- **Total Actual Deliverable**  
`SUM(Projects_Plans[Actual_delivrable])`

### 6.2 Variances

- **Cost Variance €**  
`[Total Actual Cost] - [Total Planned Cost]`
- **Cost Variance %**  
`DIVIDE([Cost Variance €], [Total Planned Cost])`
- **Duration Variance %**  
`DIVIDE([Total Actual Duration] - [Total Planned Duration], [Total Planned Duration])`
- **Deliverable Variance %**  
`DIVIDE([Total Actual Deliverable] - [Total Planned Deliverable], [Total Planned Deliverable])`

### 6.3 Alerte (seuil 15%)

- **Alert Flag (0/1 au niveau filtre/affichage)**  
Alert Flag =  
VAR Seuil = 0.15  
RETURN  
IF(  
    ABS([Cost Variance %]) >= Seuil  
    || ABS([Duration Variance %]) >= Seuil  
    || ABS([Deliverable Variance %]) >= Seuil,  
    1, 0  
)
- **Has Alert (label visible)**  
`IF([Alert Flag] = 1, "ALERTE", "OK")`

Cette logique est directement alignée avec la règle d'alerte 15% du cadrage.

## 6.4 Comptages (pilotage)

- **Alert Count** (*nb de phases en alerte dans le contexte actuel*)  
SUMX(VVALUES(Projects\_Plans[Propha\_ID]), [Alert Flag])
- **Projects In Alert** (*nb de projets distincts en alerte*)  
Projects In Alert =  
CALCULATE(  
    DISTINCTCOUNT(Projects\_Plans[Project\_ID]),  
    FILTER(VVALUES(Projects\_Plans[Project\_ID]), [Alert Flag] = 1)  
)

## 6.5 Couleurs de jauge (mise en forme conditionnelle)

- **Gauge Color** (*générique*)  
Gauge Color =  
VAR Seuil = 0.15  
VAR v = ABS([Cost Variance %])  
RETURN  
SWITCH(TRUE(),  
    v >= Seuil, "#D64541", -- rouge  
    v >= 0.10, "#F5A623", -- orange (option)  
    "#2ECC71"               -- vert  
)
- **Duration Gauge Color** : même logique basée sur [Duration Variance %]
- **Deliverable Gauge Color** : même logique basée sur [Deliverable Variance %]

---

## 7) Construction du dashboard — éléments à documenter

Le cadrage demande : vue globale filtrable + carte + info-bulles + variété de graphiques, et option Gantt/Q&A.

### 7.1 Pages (structure recommandée)

- **Main Dashboard** : synthèse + alertes + comparatifs Planned/Actual
- **Gestion de projet** : vue projet/phase (Gantt + tableau détail)

### 7.2 Slicers (interactions)

Segments utiles :

- Region, Country, Project\_type, Date, Project\_ID (selon besoins)  
Puis : **Format → Modifier les interactions** pour contrôler quels visuels sont filtrés.

### 7.3 Visuels “socle”

- Barres : *Planned vs Actual Cost / Duration / Deliverable* (par région)
- Carte monde : repérer pays avec retards/dérives (taille/couleur)  
Note+de+cadrage+Sanitoral
- Tableau : **Project\_ID + Alert Count** (priorisation)
- Jauges : synthèse global KPI + couleur conditionnelle

### 7.4 Info-bulles (Tooltips)

Créer une **page tooltip** pour afficher :

- variance %, détail pays / projets concernés
  - et l'info “sur quel KPI l'alerte porte”
- 

## 8) Procédure de mise à jour hebdomadaire (opérationnel)

C'est explicitement demandé : “procédure pour faire la mise à jour nous-mêmes”.

### Étapes

1. Remplacer/mettre à jour le fichier source **sans changer la structure** (colonnes / types)
  2. Ouvrir le **.pbix**
  3. Cliquer **Actualiser**
  4. Contrôles rapides :
    - Power Query : aucune erreur, pas de colonnes manquantes
    - 1 projet test : Planned/Actual cohérents
    - Alertes : le seuil 15% déclenche comme attendu  
Note+de+cadrage+Sanitoral
- 

## 9) Dépannage (les 4 erreurs les plus fréquentes)

- **Totaux “gonflés”** → relation N–N ou clé non unique (vérifier Propha\_ID)
- **DAX incohérent** → types mal définis (date/numérique)  
Oral P7 (1)
- **Slicers ne filtrent pas** → direction de filtre / relation inactive
- **Rôles ne filtrent pas** → vérifier relations avec User\_\* et la logique LoginUser

