

Cours Web Analyse

(GTM/GA4/Pixel/Meta)

Module 1 : Introduction à Google Tag Manager

1.1 - Qu'est-ce que Google Tag Manager (GTM) ?

Définition :

Google Tag Manager (GTM) est un outil **gratuit** de gestion des balises (tags) marketing et analytics, développé par Google. Il permet de **déployer et mettre à jour** des scripts de suivi (comme Google Analytics, Facebook Pixel, LinkedIn, etc.) **sans modifier le code source du site**.

Fonction principale :

- Centraliser tous les scripts de tracking dans une interface.
- Simplifier les mises à jour (pas besoin d'un développeur pour chaque modification).
- Améliorer la vitesse de chargement du site en évitant l'accumulation de codes manuels.

Les balises sont souvent utilisées pour :

- Suivi d'événements (Page vue, clics, formulaires, vidéos, etc.).
- Suivi des conversions (achats, leads...).
- Suivi marketing (Meta, GA4, Google Ads Remarketing).
- Tests A/B, chatbots, heatmaps (Hotjar, Clarity).

1.2 - Pourquoi utiliser GTM plutôt que le code manuel ?

Avantages de GTM	Problèmes du code manuel
✓ Pas besoin de modifier le code source (idéal pour les non-développeurs).	✗ Requier l'intervention d'un dev pour chaque modification.
✓ Déploiement rapide des balises (temps réduit de A/B testing).	✗ Risque d'erreurs (typos, mauvais emplacements).
✓ Gestion centralisée (tous les tags dans une seule interface).	✗ Accumulation de scripts = ralentissement du site.
✓ Versionning (historique des changements et rollback possible).	✗ Difficile à auditer (qui a ajouté quel code ?).
✓ Debugging intégré (mode prévisualisation pour tester avant publication).	✗ Débogage complexe (nécessite la console navigateur).

Cas d'usage idéal pour GTM :

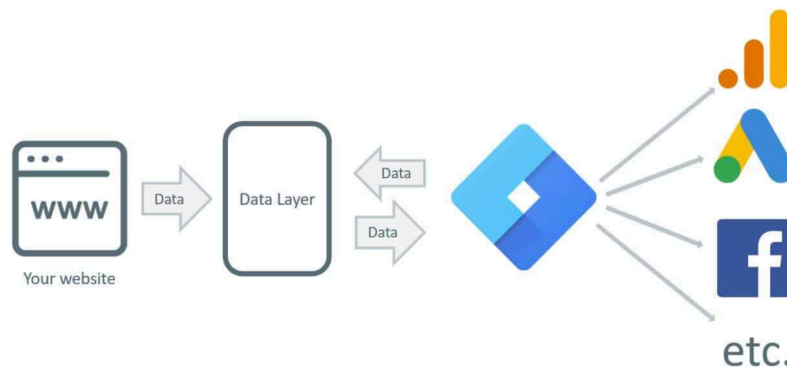
- Équipes marketing/managers qui veulent **autonomie**.
- Sites avec **multiples outils de tracking**.
- Environnements où les mises à jour de code sont **longues ou coûteuses**.

1.3 - Différences entre GTM et Google Analytics :

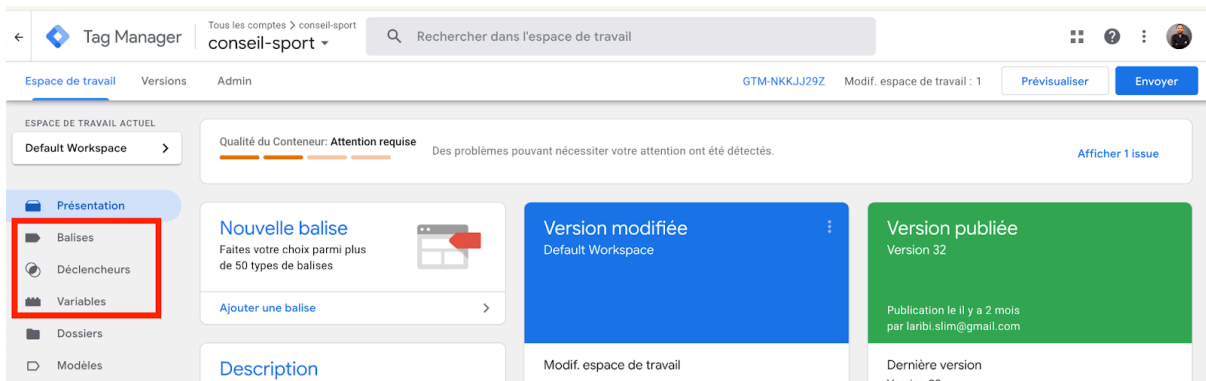
Google Tag Manager (GTM)	Google Analytics (GA4)
• Outil de gestion des balises.	• Outil de mesure et analyse des données.
• Permet d'installer d'autres outils (Facebook Pixel, Google Ads, etc.).	• Se limite au suivi Google (GA4).
• Ne stocke pas de données.	• Collecte et stocke les données utilisateurs.
• Interface axée sur les tags/triggers/variables.	• Interface axée sur les rapports et insights.

1.4 - Architecture et fonctionnement de GTM

Schéma de flux :



Composants clés de GTM :



1. **Conteneur** : Fichier JavaScript injecté sur le site (contient tous les tags).
2. **Tags (Balise)** : Scripts à déclencher (ex: "Envoyer un événement à GA4").
3. **Triggers (Déclencheur)** : Règles de déclenchement (ex: "Quand un clic survient sur le bouton A").
4. **Variables** : Données dynamiques (ex: URL de la page, ID de produit).

1.5 - Création d'un compte et d'un conteneur

Étape 1 : Créer un compte GTM

1. Aller sur tagmanager.google.com.
2. Cliquer sur "Créer un compte".
3. Remplir :
 - **Nom du compte** (ex: "Mon Entreprise").
 - **Pays**.
 - **Nom du conteneur** (ex: "Site Web Principal").
 - **Plateforme** (Web, iOS, Android, AMP, Server).

← Ajouter un compte

Configuration du compte

Nom du compte

par exemple, Mon entreprise

Pays

États-Unis

☐ Partager des données anonymement avec Google et des tiers

Configuration du conteneur

Nom du conteneur

Exemple : www.mysite.com

Plate-forme cible

- Web**
À utiliser sur les pages Web mobiles et pour ordinateur
- iOS**
À utiliser dans les applications iOS
- Android**
À utiliser dans les applications Android
- AMP**
À utiliser dans les pages AMP (Accelerated Mobile Pag...
- Server**
For server-side instrumentation and measurement

Étape 2 : Installer le conteneur sur le site Wordpress

Après création, GTM fournit un code à 2 parties :

- **<head>** : Charge le script GTM.
- **<body>** : Gère les balises si JavaScript est désactivé.

Installer Google Tag Manager

Copiez le code ci-dessous et collez-le sur toutes les pages de votre site Web.

1. Collez ce code le plus haut possible dans la section **<head>** de la page :

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
  new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
  j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
  'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
  })(window,document,'script','dataLayer','GTM-NKKJJ29Z');</script>
<!-- End Google Tag Manager -->
```

2. Collez ce code juste après la balise d'ouverture **<body>** :

```
<!-- Google Tag Manager (noscript) -->
<noscript><iframe src="https://www.googletagmanager.com/ns.html?id=GTM-NKKJJ29Z"
  height="0" width="0" style="display:none;visibility:hidden"></iframe></noscript>
<!-- End Google Tag Manager (noscript) -->
```

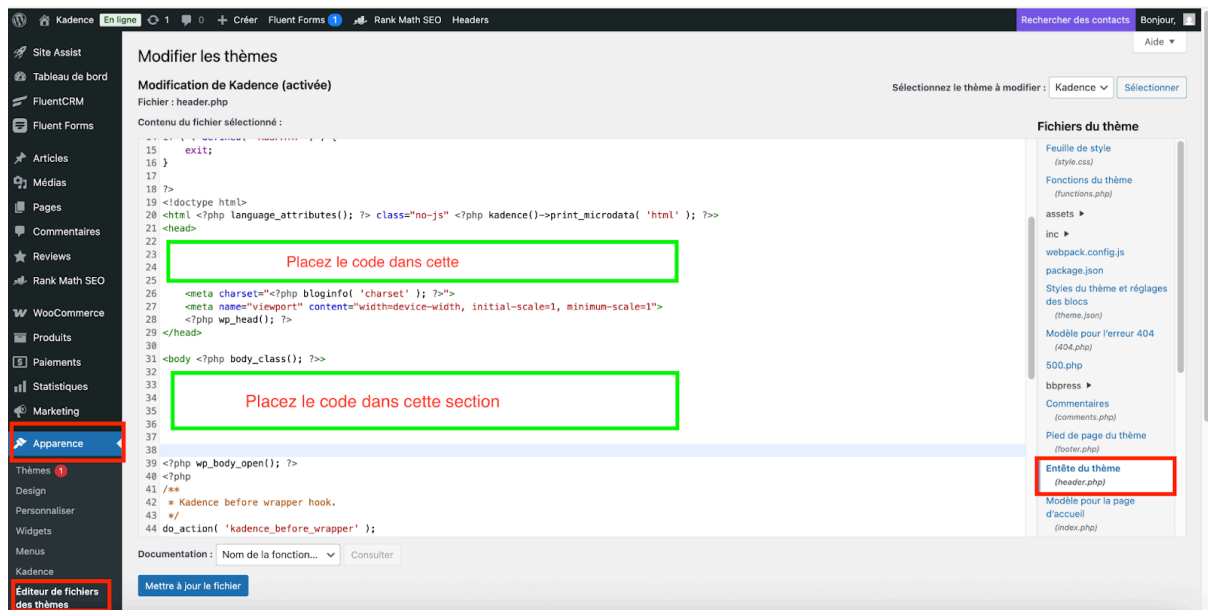
3. Testez votre site Web (facultatif) :

Exemple :

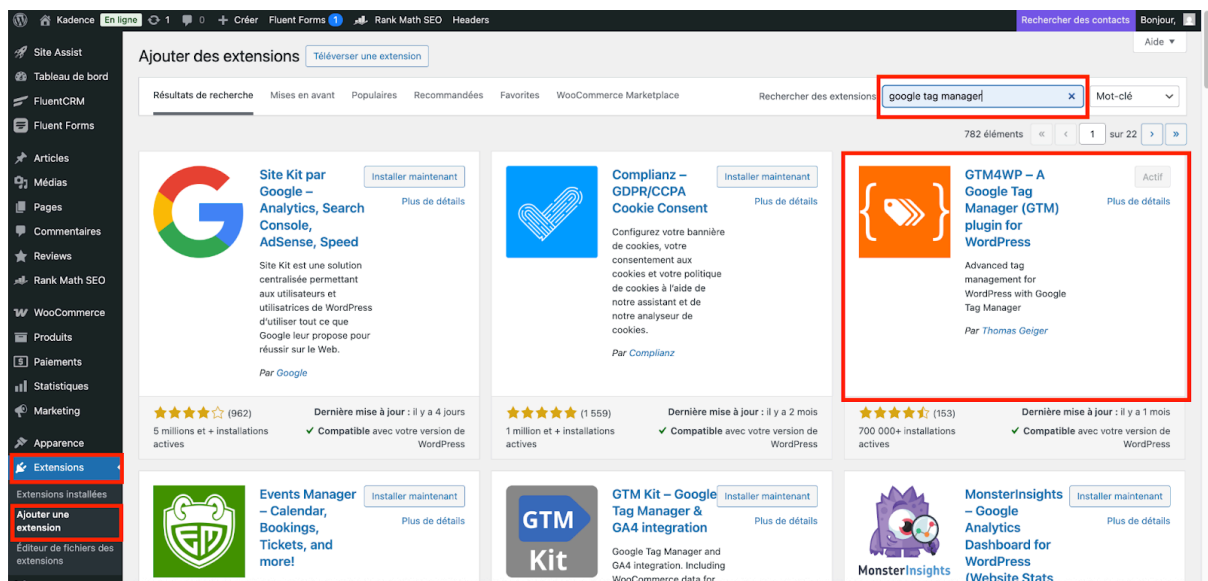
Pour en savoir plus sur l'installation de l'extrait Google Tag Manager, consultez notre [guide de démarrage rapide](#).

Méthodes d'installation :

- **Manuelle** : Coller le code dans le **<head>** du site.



- CMS : Plugins pour WordPress (ex: "Google Tag Manager for WordPress").



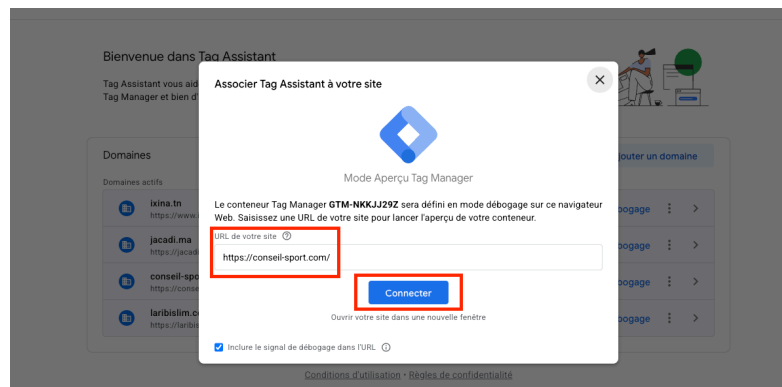
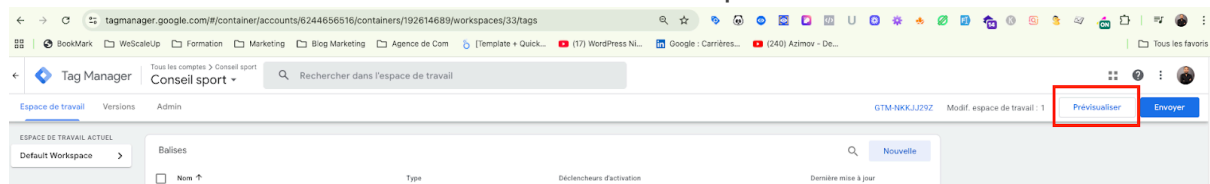
Étape 3 : Vérifier l'installation (2 Méthodes)

- Utiliser l'extension **Google Tag Assistant** (Chrome).

<https://chromewebstore.google.com/detail/tag-assistant/kejbjndbnbjgmefkgdddjlbokphdefk>



- Activer le **Mode Prévisualisation** dans GTM pour tester en direct.

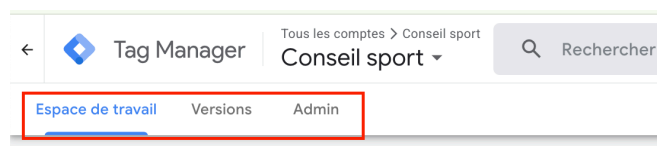


Module 2 : Interface et Configuration de Base de Google Tag Manager

2.1 - Présentation de l'interface GTM

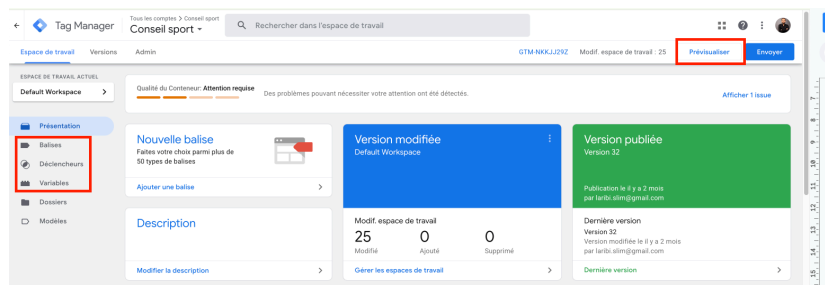
L'interface GTM est divisée en **4 sections principales** :

1. **Menu de Navigation** (à gauche) :
 - **Workspace** : Espace de travail actif (tags/triggers/variables).
 - **Versions** : Historique des modifications publiées.
 - **Admin** : Gestion des utilisateurs et paramètres.



2. **Tableau de Bord** :
 - Aperçu des **tags**, **triggers**, et **variables** existants.

- Bouton "**Prévisualiser**" pour le mode debug.



3. **Composants Principaux** (Tags, Triggers, Variables) :
 - **Tags** : Balises à déclencher (ex: GA4, Facebook Pixel).
 - **Triggers** : Règles de déclenchement (ex: clic, page vue).
 - **Variables** : Données dynamiques utilisées par les tags/triggers.
4. **Barre d'Actions** :
 - Boutons "**Envoyer**" (publier) et "**Annuler**" (revenir en arrière).

2.2 - Comprendre les Composants Principaux

A. Tags (Balises)

Définition : Scripts exécutés sur le site (ex: Google Analytics, Pixel FB, LinkedIn Insight, etc)).

Exemples de Tags natifs dans GTM :

- **Google Analytics : GA4 Configuration / GA4 Event.**
- **Google Ads Conversion Tracking.**
- **HTML personnalisé** (pour scripts externes).

☐	GA4 - Event - Page 404	Google Analytics : Événement GA4	 Page 404
☐	GA4 - Event - Purchase	Google Analytics : Événement GA4	 purchase
☐	GA4 - Event - Scrolling	Google Analytics : Événement GA4	 Scroll
☐	GA4 - Event - view_item	Google Analytics : Événement GA4	 view_item

Bonne pratique de nomenclature (standard pro)

La logique à suivre est :

[Outil] – [Type] – [Nom de l'action]

👉 Pour GA4 ça donne :

- **GA4 – Config – [Nom du flux]**
- **GA4 – Event – Page_view**
- **GA4 – Event – View_item**
- **GA4 – Event – Purchase**

- GA4 – Event – Scroll
- GA4 – Event – Click – WhatsApp
- GA4 – Event – Click – PDF

B. Triggers (Déclencheurs)

Définition : Conditions pour lancer un tag.

Triggers intégrés courants :

- **Page View** : Déclenché au chargement de la page.
- **Clic** : Sur un élément spécifique (bouton, lien).
- **Form Submission** : Après soumission d'un formulaire.

	Page vue
	Clic
	Liens uniquement
	Tous les éléments
	Engagement utilisateur
	Envoi de formulaire
	Profondeur de défilement
	Vidéo YouTube

C. Variables

Définition : Stockent des données dynamiques pour les tags/triggers.

Types de Variables :

- **Built-in Variables** (prédéfinies) : Page URL, Click Text, etc.
- **Variables personnalisées** : Via Data Layer DLV ou JavaScript.

Variables définies par l'utilisateur

☐ Nom ↑	Type
☐ 404 not found	Variable JavaScript
☐ DLV - content_category	Variable de couche de données
☐ DLV - content_ids	Variable de couche de données
☐ DLV - content_type	Variable de couche de données
☐ DLV - contents	Variable de couche de données
☐ DLV - currency	Variable de couche de données

Module 3 : Les Balises (Tags) dans GTM

3.1 - Types de balises prises en charge

GTM prend en charge des dizaines de balises natives et personnalisées. Voici les principales catégories :

Balises natives (intégrées à GTM)

	Exemples d'utilisation
Google Analytics : GA4	Suivi de page views, événements
Google Ads Conversion	Suivi des conversions publicitaires
Facebook Pixel	Remarketing et suivi des conversions
LinkedIn Insight Tag	Suivi des campagnes LinkedIn
TikTok Pixel	Suivi des conversions TikTok Ads
YouTube Embedded Player	Suivi des interactions avec les vidéos

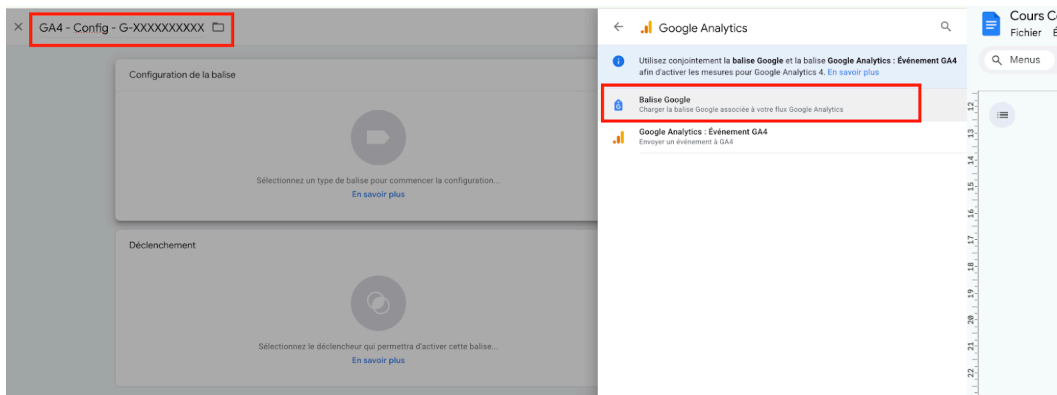
Balises personnalisées

- **HTML personnalisé** : Pour tout script non natif
 - **JavaScript personnalisé** : Pour des fonctions avancées
-

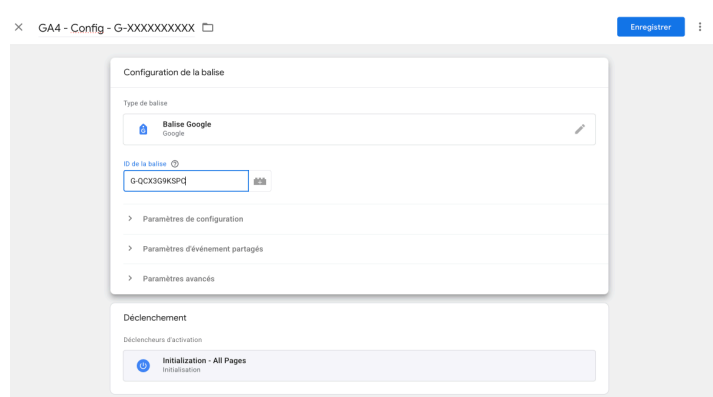
3.2 - Configuration d'un tag Google Analytics 4 (GA4)

Étapes pour créer un tag GA4

1. **Créer un nouveau tag**
 - Dans GTM → Tags → Nouveau
 - Nommer le tag (ex: "GA4 - Config - ")
2. **Choisir le type de tag**
 - Sélectionner "**Google Analytics : GA4 Configuration**"
3. **Remplir les paramètres**
 - **ID de mesure GA4** (ex: G-XXXXXXXXXX) → Trouvable dans GA4



- (Optionnel) **Paramètres avancés** comme le délai de session
- 4. **Définir le trigger**
 - Associer le trigger "**All Pages**" pour un suivi global



5. Enregistrer et publier

3.3 Dimensions et paramètres d'événements GA4

En GA4, chaque balise peut transmettre des **dimensions** (informations contextuelles) sous forme de **paramètres d'événement**.

Voici les principales dimensions utiles, leur rôle, et pour quelles balises les utiliser :

Navigation & Page View

- **page_location** → URL complète de la page ({{Page URL}})
- **page_referrer** → Page précédente ({{Referrer}})
- **page_title** → Titre de la page ({{Page Title}})
- **language** → Langue du navigateur de l'utilisateur ({{Language}})
- **screen_resolution** → Résolution d'écran ({{Screen Resolution}})

👉 Balises concernées :

- **GA4 – Config – Main** (automatique)

Scroll Tracking

- **percent_scrolled** → Niveau de scroll atteint ({{Scroll Depth Threshold}})

👉 Balises concernées :

- **GA4 – Event – Scroll**
-

3. Click Tracking

- **button_text** → Texte du bouton ({{Click Text}})
- **button_id** → ID HTML du bouton ({{Click ID}})
- **button_classes** → Classe CSS ({{Click Classes}})
- **button_url** → Lien du bouton ({{Click URL}})

👉 Balises concernées :

- **GA4 – Event – Click_button**
 - **GA4 – Event – Click – WhatsApp / PDF /**
-

4. Site Search (barre de recherche)

- **search_term** → Mot-clé tapé dans la recherche interne ({{Query Variable}})

👉 Balises concernées :

- **GA4 – Event – Search**
-

6. Formulaires & Leads

- **form_id** → ID du formulaire (ex. contact_form)
- **form_name** → Nom du formulaire (ex. Formulaire Newsletter)
- **method** → Méthode d'inscription ou de login (Google, Email, Facebook)

👉 Balises concernées :

- **GA4 – Event – generate_lead**
- **GA4 – Event – Sign_up**
- **GA4 – Event – Login**

Dimension / Paramètre	Exemple	Utilisation	Balise concernée
page_location	https://monsite.com/home	Connaître l'URL visitée	Page_view
page_referrer	https://google.com	Connaître la source précédente	Page_view
page_title	Chaussures running	Identifier le titre SEO	Page_view
percent_scrolled	50	Mesurer profondeur de scroll	Scroll
button_text	"Acheter"	Identifier le bouton cliqué	Click_button
search_term	"chaussures"	Requêtes recherche interne	Search
transaction_id	CMD12345	Suivi des commandes	Purchase
value	249.99	Montant de la commande	Purchase
form_id	contact_form	Identifier le formulaire soumis	Generate_lead
video_title	Promo 2025	Suivi d'engagement vidéo	Video_progress

Pourquoi respecter les noms prédéfinis (événements recommandés) ?

Lien :

<https://support.google.com/analytics/answer/9322688?hl=fr&sjid=6133492794513964219-EU#zippy=%2Crealtime-report%2Cdebugview-report%2Crappport-temps-r%C3%A9el%2Crappport-debugview>

Google a défini une **liste d'événements standards** dans GA4 (ex : **purchase**, **add_to_cart**, **sign_up**, **generate_lead**).

👉 Si tu utilises ces noms exacts :

1. GA4 les reconnaît automatiquement

- Tu auras des rapports prêts à l'emploi (pas besoin de tout reconstruire en explorations).
- Exemple : **purchase** → GA4 le met direct dans les rapports de revenus.

2. Compatibilité avec Google Ads & remarketing

- Google Ads comprend que **generate_lead** = une conversion lead.

◆ Exemple concret

👉 Si tu crées un event :

- **Nom** : **generate_lead**
- **Paramètres** : **form_id**, **form_name**, **method**

Quand quelqu'un soumet un formulaire de contact :

- **event_name**: "generate_lead"
- **form_id**: "contact_form"
- **form_name**: "Formulaire Contact"
- **method**: "email"

Paramètre	Valeur dans GTM	Explication
form_id	{{Form ID}}	Identifiant unique du formulaire (si présent dans le code HTML → attribut id=)
form_name	{{Form Classes}} ou variable personnalisée	Nom ou classe du formulaire (ex : newsletter_form , contact_form)
method	"contact" / "newsletter" / "devis" (valeur fixe)	Le type de lead ou méthode (tu peux mettre une valeur fixe en texte si tu sais d'où ça vient)

page_location	{{Page URL}}	Page où le lead est généré
page_title	{{Page Title}}	Titre de la page

Tu peux le laisser vide si c'est juste un formulaire de contact classique.

json

 Copier le code

```
event_name: "generate_lead",
form_id: "contact_form",
form_name: "Formulaire de Contact",
method: "contact",
page_location: "https://www.monsite.com/contact",
page_title: "Contact",
value: 10,
currency: "TND"
```

 GA4 le traite comme un vrai lead, visible dans **Conversions** et exportable vers Google Ads.

◆ Et si tu mets **formsubmit** ou **Lead** ?

- GA4 ne va **pas bloquer l'événement** → il apparaîtra quand même.
- Mais **✗** tu n'auras pas les rapports standardisés.
- **✗** Google Ads ne saura pas automatiquement que c'est un "lead".
- Tu devras tout reconstruire en **événement personnalisé** → plus de travail, moins de compatibilité

En GA4, il y a une différence entre **événements recommandés** et **événements personnalisés** :

Cas d'usage recommandé : **generate_lead**

Google a défini l'événement **generate_lead** pour toutes les situations où un utilisateur remplit un formulaire qui génère une opportunité business (ex : demande de devis, inscription newsletter, prise de RDV...).

 Pourquoi l'utiliser ?

- GA4 le reconnaît comme **conversion standard**.
- Tu le retrouves facilement dans les rapports.
- Tu peux l'envoyer vers **Google Ads** et l'utiliser pour optimiser tes campagnes.
- Tu bénéficies d'une **comparabilité** : toutes les plateformes qui utilisent GA4 parlent le même langage (lead = generate_lead).

Et si j'ai plusieurs formulaires (lead vs simple contact) ?

Tu n'as pas besoin de créer deux noms d'événements différents.

👉 Utilise **toujours generate_lead**, mais fais varier les **paramètres** :

Exemple :

- **Formulaire devis / demande d'offre**

```
event_name: "generate_lead",
form_id: "devis_form",
form_name: "Formulaire Devis",
method: "contact"
```

- **Formulaire simple de contact :**

```
event_name: "generate_lead",
form_id: "contact_form",
form_name: "Formulaire Contact",
method: "contact"
```

Tu auras donc un seul event (**generate_lead**), mais tu différencies tes leads grâce aux paramètres **form_id** ou **form_name**.

Quand créer un autre event personnalisé ?

Seulement si ce n'est **pas un vrai lead**.

Exemple : un formulaire de feedback ou un quiz → là, tu pourrais créer un event custom type **form_submit** pour ne pas polluer tes stats de leads.

Exemple de code d'un formulaire WordPress (Elementor, Contact Form 7, etc.) :

html

 Copier le code

```
<form id="contact_form" class="formulaire-contact" name="contact">
  <input type="text" name="nom" />
  <input type="email" name="email" />
  <button type="submit">Envoyer</button>
</form>
```

Paramètre d'événement	Valeur GTM	Exemple transmis
form_id	{{Form ID}}	"contact_form"
form_name	{{Form Classes}}	"formulaire-contact"
method	{{Form Target}} ou {{Page URL}} ou valeur fixe	"contact"

👉 Ces variables GTM vont automatiquement lire les attributs du formulaire HTML de WordPress.

✅ Conclusion :

Il faut **toujours respecter les noms d'événements prédéfinis de GA4** (`generate_lead`, `sign_up`, `purchase`, etc.), sauf si tu as un cas très spécifique.

Pour tes 2 formulaires (**lead et contact simple**) → utilise **generate_lead**.

Distingue-les grâce à `form_id` / `form_name`.

Ne crée un event personnalisé que si ce n'est pas un lead exploitable.

3.3 - Ajout de balises personnalisées (HTML/JavaScript)

Cas d'usage typiques

- Intégration d'outils non natifs (ex: Hotjar, outils CRM)
- Scripts de personnalisation avancée

Méthode pas à pas

1. **Créer un nouveau tag** → **HTML personnalisé**
2. **Coller le code** fourni par l'outil tiers



```
html                                                                    Copy

<!-- Exemple : Pixel Hotjar -->
<script>
  (function(h,o,t,j,a,r){
    h.hj=h.hj||function(){(h.hj.q=h.hj.q||[]).push(arguments)};
    h._hjSettings={hjid:1234567,hjsv:6};
    a=o.getElementsByTagName('head')[0];
    r=o.createElement('script');r.async=1;
    r.src=t+h._hjSettings.hjid+j+h._hjSettings.hjsv;
    a.appendChild(r);
  })(window,document,'https://static.hotjar.com/c/hotjar-','.js?sv=');
</script>

▶ Run HTML
```

3. **Définir le trigger** (ex: "All Pages")
4. **Tester en prévisualisation** avant publication

⚠ Bonnes pratiques

- Limiter les balises personnalisées pour éviter les conflits
- Utiliser des commentaires dans le code pour documentation

3.4 - Gestion des balises tierces (LinkedIn, TikTok, etc.)

Méthode standard pour les pixels

1. **Récupérer le code du pixel** depuis la plateforme (ex: LinkedIn Ads)
2. **Créer un tag HTML personnalisé** dans GTM
3. **Configurer le trigger** (ex: "Page View" ou événement spécifique)

Exemple : Configuration du Pixel LinkedIn

1. Dans LinkedIn Campaign Manager → **Créer un pixel**
2. Dans GTM :
 - Type de tag : **HTML personnalisé**

- Coller le script LinkedIn
- Trigger : **All Pages** (ou une page de conversion spécifique)

Optimisation avancée

- **Utiliser des variables** pour dynamiser les paramètres (ex: {{Page URL}})
- **Ordonnancer les tags** pour prioriser les pixels critiques

Module 4 : Les Triggers (Déclencheurs) dans GTM

4.1 - Comprendre les déclencheurs natifs

Qu'est-ce qu'un trigger ?

Un trigger (déclencheur) définit **quand et où** un tag doit s'activer. Sans trigger, un tag ne fonctionnera jamais.

Triggers intégrés principaux

Type de Trigger	Utilisation typique	Exemple d'application
Page View	Déclencher un tag au chargement de page	Installation de GA4 sur toutes les pages
Clic	Suivre les interactions avec éléments	Clics sur "Ajouter au panier"
Form Submission	Capturer les soumissions de formulaires	Suivi des leads
Timer	Déclencher après un délai spécifique	Pop-up après 30 secondes
Scroll Depth	Suivre le défilement de page	Mesure d'engagement

4.2 - Créer des déclencheurs personnalisés

Étapes pour créer un trigger

1. Dans GTM → **Triggers** → **Nouveau**
2. Nommer le trigger (ex: "Clic - Bouton Contact")
3. Choisir le type de trigger
4. Configurer les paramètres

Exemple : Trigger de clic avancé

plaintext

Copy

```
Configuration :  
- Type : Clic – Juste les liens  
- Déclencher sur : Certains clics  
- Conditions : {{Click URL}} contient "contactez-nous"
```

4.3 - Utilisation des événements courants

A. Suivi des clics

1. Activer la variable **Click Variables** dans GTM
2. Créer un trigger de type **Clic**
3. Configurer :
 - **Juste les liens** ou **Tous les éléments**
 - Filtres optionnels (ID, classe CSS, URL)

B. Scroll Depth (profondeur de défilement)

1. Créer un trigger **Scroll Depth**
2. Paramétrer :
 - Seuils (25%, 50%, 75%, 90%)
 - Horizontal/Vertical

C. Formulaires

plaintext

Copy

```
Configuration type :  
- Type : Form Submission  
- Attendre : Validation réussie  
- Vérifier : {{Form ID}} = "newsletter-form"
```

4.4 - Déclencheurs basés sur le temps (Timer)

Cas pratique : Pop-up d'abandon

1. Créer un trigger **Timer**
2. Paramètres :
 - Intervalle : 30000 ms (30 sec)
 - Limite : 1 fois par page
3. Associer à un tag HTML (code du pop-up)

Optimisation

- Combiner avec **Page Path** pour ne pas déclencher sur certaines pages
- Utiliser **Window Loaded** comme condition supplémentaire

Bonnes pratiques des triggers

1. **Nommage clair** : "Clic - Bouton Principal (Header)"
2. **Tests systématiques** : Mode prévisualisation avant publication
3. **Ordonnancement** : Définir l'ordre d'exécution si nécessaire
4. **Exceptions** : Exclure des pages/éléments si besoin

Module 5 : Les Variables dans GTM

5.1 - Rôle des Variables dans GTM

À quoi servent les variables ?

Les variables dans GTM sont des **conteneurs dynamiques** qui stockent des données utilisables par :

- **Les Tags** (ex: envoyer l'URL actuelle à GA4)
- **Les Triggers** (ex: déclencher un tag seulement si `{{Page Path}}` = `/checkout`)
- **Les Variables elles-mêmes** (imbrication possible)

Exemple concret

plaintext

Copy

```
Tag Facebook Pixel :  
- Paramètre "page_url" = {{Page URL}}  
- Paramètre "user_id" = {{User ID}} (variable personnalisée)
```

5.2 - Variables intégrées (Built-in Variables)

Principales variables natives

Catégorie	Variables utiles	Exemple d'usage
Page	Page URL, Page Path	Filtrage par URL dans les triggers
Clics	Click Text, Click URL	Suivi des boutons spécifiques
Formulaires	Form ID, Form Text	Validation de champ
Divers	Scroll Depth, Video Status	Suivi d'engagement

Comment les activer ?

1. Aller dans **Variables**

2. Cliquer sur "**Configurer**"
3. Cochez les cases des variables nécessaires

Les variables de clic (Click Variables)

Pourquoi tracker les clics ?

Un clic est une **action utilisateur clé** :

- clic sur "Acheter"
- clic sur "Ajouter au panier"
- clic sur "Contact"
- clic sur un bouton WhatsApp
- clic sur un CTA marketing

👉 Sans tracking de clic :

- on ne sait pas **ce qui attire l'utilisateur**
- on ne sait pas **quel bouton convertit**
- on ne peut pas **optimiser l'UX et le tunnel de conversion**

Google Tag Manager permet de **transformer un clic en donnée mesurable**.

Comment GTM "voit" un clic ?

Quand un utilisateur clique sur un élément, GTM peut lire les **attributs HTML** de cet élément.

Exemple de bouton HTML :

```
<button id="buy_btn" class="btn btn-primary">Acheter</button>
```

GTM peut récupérer :

- l'**ID** → `buy_btn`
- les **classes CSS** → `btn btn-primary`
- le **texte visible** → `Acheter`

Ces informations sont appelées **variables de clic**.

5.3 - Variables personnalisées

Types de variables personnalisées

1. **Variable Data Layer**
 - Récupère des données du `dataLayer.push()`
 - Exemple :

javascript

Copy

```
dataLayer.push({
  'event': 'purchase',
  'transactionValue': 100
});
```

- → Créez une variable transactionValue pour l'utiliser dans un tag GA4.

2. Variable JavaScript

- Exécute du code JS pour extraire des données
- Exemple (récupérer la classe CSS d'un élément) :

javascript

Copy

```
function() {
  return document.querySelector('.price').innerText;
}
```

3.:Variable Cookie

- Lit un cookie spécifique (ex: _ga pour l'ID client GA)

Cas pratique : Récupérer un ID produit

1. Créez une variable JavaScript :

javascript

Copy

```
function() {
  return document.querySelector('.product-id').value;
}
```

2. Utilisez-la dans un tag GA4 comme paramètre item_id.

5.4 - Utilisation avancée du Data Layer

Best Practices

- **Nommage cohérent :**

javascript

Copy

```
dataLayer.push({
  'event': 'login',
  'userType': 'premium'
});
```

- **Variables imbriquées :**
Créez une variable userType pour réutiliser la valeur ailleurs.

Debugging

- **Extension Chrome "Data Layer Inspector"**
- **Console navigateur :** Tapez dataLayer pour voir son contenu

Module 6 : Suivi Avancé avec GTM

(Applications pratiques avec variables et Data Layer)

6.1 - Implémentation du Enhanced Ecommerce (GA4)

Étapes pour le suivi e-commerce

1. Configurer le Data Layer dans le code du site :

```
javascript Copy
dataLayer.push({
  'event': 'view_item',
  'ecommerce': {
    'items': [{
      'item_id': 'SKU123',
      'item_name': 'T-Shirt',
      'price': 29.99,
      'item_category': 'Vêtements'
    }]
  }
});
```

2. Créer des variables dans GTM pour chaque champ :
 - ecommerce.items.0.item_id
 - ecommerce.items.0.price
3. Configurer un tag GA4 :
 - Type : **GA4 Event**
 - Nom de l'événement : view_item
 - Paramètres : Mappez les variables du Data Layer
4. **Trigger** : Événement personnalisé view_item

Bonnes pratiques :

- Vérifiez la structure des données avec **Preview Mode**
- Utilisez **Google Analytics Debugger** (extension Chrome)

6.2 - Tracking des vidéos YouTube

Méthode pour suivre les lectures

1. Activer le trigger natif :
 - Dans GTM → Triggers → **YouTube Video**
 - Options :
 - Début/Fin/Pause/Progression à 25%, 50%, 75%
2. Créer un tag GA4 :

- Événement : video_progress
- Paramètres :
 - video_title : {{YouTube Video Title}}
 - progress : {{YouTube Video Percent}}

3. Exemple de données envoyées :

```
json
{
  "video_title": "Tutoriel GTM",
  "progress": "50%"
}
```

6.3 - Suivi des téléchargements (PDF, ZIP)

Configuration

1. Créer un trigger :
 - Type : **Clic - Juste les liens**
 - Condition : {{Click URL}} se termine par .pdf ou .zip
2. Tag GA4 :
 - Événement : file_download
 - Paramètres :
 - file_name : {{Click URL}}
 - file_type : {{Click Extension}} (variable personnalisée)

6.4 - Mesure des interactions utilisateurs

A. Clics hors page (Outbound Links)

1. Trigger :
 - Condition : {{Click URL}} commence par "http" ET ne contient pas "votresite.com"
2. Tag :
 - Événement : outbound_click
 - Paramètres :
 - link_url : {{Click URL}}

B. Affichage de pop-ups

1. Data Layer lors de l'ouverture :

```
javascript
dataLayer.push({
  'event': 'popup_view',
  'popup_id': 'newsletter_modal'
});
```

2. Tag GA4 associé à l'événement personnalisé popup_view.

6.4 - Tracking Page 404

C'est quoi une page 404 ?

Une **page 404** apparaît lorsqu'un utilisateur essaie d'accéder à une page qui **n'existe pas** ou **plus** :

- lien cassé
- produit supprimé
- URL mal tapée
- ancienne page indexée par Google

👉 Le serveur répond : **"Page introuvable"** ou **"Page Not Found"**

Pourquoi GA4 ne tracke pas les 404 automatiquement ?

GA4 voit une page 404 comme **une page normale** :

- même URL
- même page_view
- aucun événement spécifique

👉 Sans configuration, **tu ne sais pas** :

- combien de 404 tu as
- d'où viennent les utilisateurs
- quelles pages cassent l'expérience

Principe du tracking page 404 (GTM) :

👉 Objectif : **Envoyer un événement GA4 uniquement quand la page affichée est une vraie 404**

Problème

- L'URL ne contient pas toujours **/404**
- GA4 ne détecte rien automatiquement

Solution

On s'appuie sur :

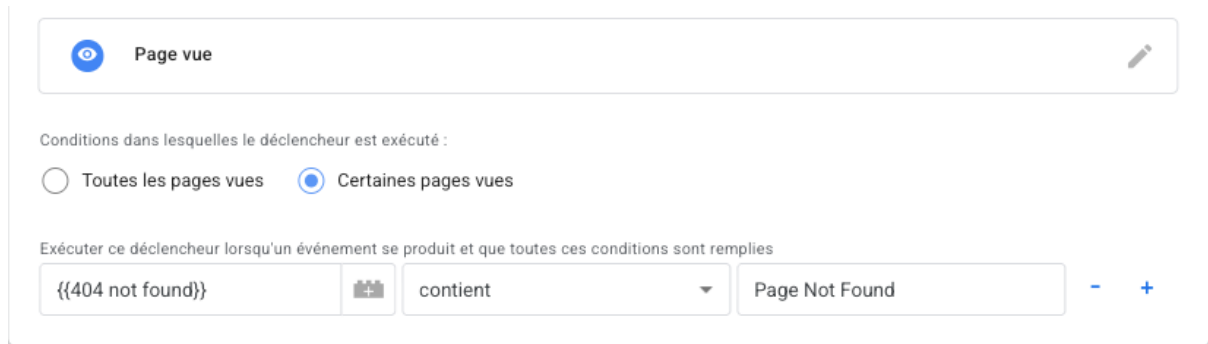
- le **title de la page**
- des **conditions de sécurité**

Méthode simple et fiable (document.title):

Étape 1 – Variable GTM

Créer une variable JavaScript : **document.title**

Étape 2 – Déclencheur :



The screenshot shows the GTM 'Page vue' (Page View) trigger configuration. At the top, there's a header with a blue eye icon and the text 'Page vue'. Below this, a section titled 'Conditions dans lesquelles le déclencheur est exécuté :' (Conditions in which the trigger is executed) contains two radio buttons: 'Toutes les pages vues' (All pages viewed) and 'Certaines pages vues' (Some pages viewed), with the latter being selected. A sub-header reads 'Exécuter ce déclencheur lorsqu'un événement se produit et que toutes ces conditions sont remplies' (Execute this trigger when an event occurs and all these conditions are met). Below this, a list of conditions is shown: the first condition is '{{404 not found}}' (selected with a blue icon), followed by 'contient' (contains) in a dropdown menu, and 'Page Not Found' as the value. To the right of the conditions are minus and plus icons for removing or adding more conditions.

Étape 3 – Sécurité supplémentaire (recommandée)

On ajoute une 2e condition : **(Pour éviter de détecter à tort des articles de blog qui parlent des pages 404, alors qu'il ne s'agit pas de vraies pages d'erreur.)**

- Page URL ne contient pas /blog/

Double validation = tracking propre

Enhanced Ecommerce Tracking

Balise View_item

Envoyer à GA4 l'événement **view_item** avec les infos produit (id, nom, prix, etc.) à chaque ouverture de **page produit**.

Pourquoi ? → Pour avoir un **reporting e-commerce fiable** (impressions produit, vues, clics, ajout panier, achat...) et nourrir **Google Ads**.

Prérequis (1 minute)

- Un **Property GA4** (ID type **G-XXXXXXX**)
- Un **container GTM** installé sur le site
- **Une seule balise** "GA4 – Config – Main" déjà créée dans GTM (avec ton Measurement ID)

Pourquoi ? → La balise Config envoie le flux de base (page_view, session...); toutes les balises Event s'appuient dessus.

Étape 1 — Faire apparaître les données produit dans le dataLayer

Deux façons (choisis 1) :

Option A — Plugin WordPress (simple) : *GTM4WP*

1. Installe/active **GTM4WP**
2. Active **Enhanced Ecommerce (GA4)** dans ses réglages
→ Le plugin poussera automatiquement un `dataLayer.push({ event: "view_item", ecommerce: {...} })` sur les fiches produit.

Pourquoi ? → Zéro code, standard propre.

Option B — Code (souple) : WPSnippet / thème / PrestaShop

Ajoute sur la page produit un **dataLayer.push** similaire à :

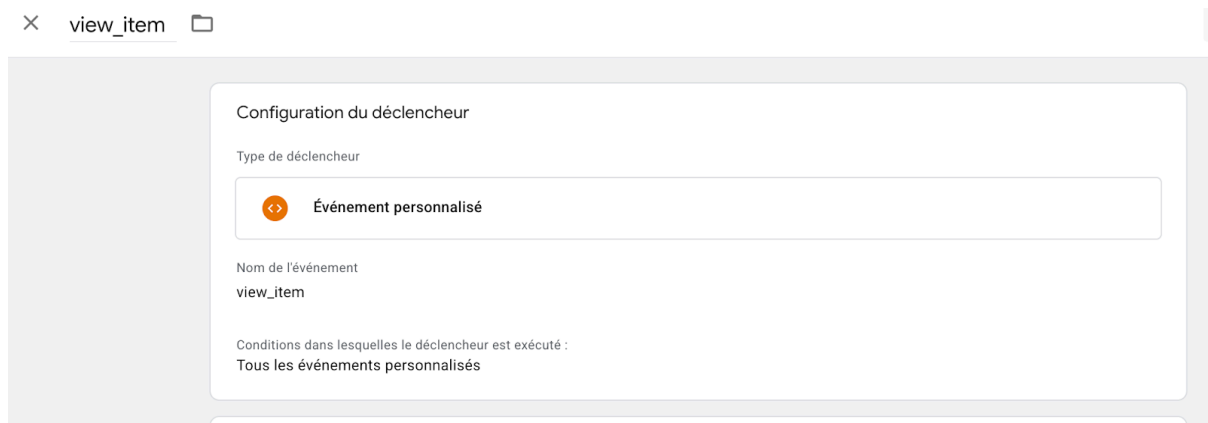
```
<script>
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({
  event: "view_item",
```

```
ecommerce: {
  currency: "TND",
  // value est optionnel pour view_item
  items: [{
    item_id: "SKU123",
    item_name: "Chaussures de running homme",
    item_category: "Chaussures",
    price: 199.99,
    quantity: 1
  }]
}
});
</script>
```

Pourquoi ? → GA4 ne “devine” pas tes produits. Tu dois lui **donner** l’objet `ecommerce` (items, prix, devise...).

Étape 2 — Créer le déclencheur (Trigger) dans GTM

- Type : **Custom Event**
- **Event name** : `view_item` (tout en minuscules, **underscore**, pas de tiret)
- Portée : **All Custom Events** (ou limite-le aux pages produit si tu veux)



Pourquoi ? → Le déclencheur écoute le **même nom d'événement** que celui poussé dans le dataLayer.

Étape 4 — Créer la balise GA4 – Event – view_item

- Type : **Google Analytics: GA4 Event**
- **Event Name** : `view_item`

- **Configuration Tag** : GA4 – Config – Main
- **Ecommerce** : coche **Send Ecommerce data** → Data Layer
- **Ne rien ajouter** dans “Event Parameters” pour **items**, **value**, **currency** si tu coches “Send Ecommerce data”

GA4 - Event - view_item

The screenshot shows the GA4 Event configuration for 'view_item'. At the top, the container ID is 'G-42GNS57BVW'. The event name is 'view_item'. Under 'Paramètres de l'événement', the 'Event Settings Variable' is set to 'Aucun'. A table lists event parameters: 'currency' with value '{{DLV - currency}}', 'items' with value '{{DLV - items}}', and 'value' with value '{{DLV - value}}'. A red box highlights this table. Below, under 'Plus de paramètres', the 'E-commerce' section is expanded, showing a checkbox for 'Envoyer des données d'e-commerce' which is currently unchecked. Another red box highlights this checkbox.

Pourquoi ?

- L'option “Send Ecommerce data” dit à GTM : “prends l’objet **ecommerce** du **dataLayer** et envoie-le tel quel”.
- Éviter de **dupliquer** les mêmes infos en Event Parameters (sinon risques d’écrasement/doublons).

Étape 7 — Bonnes pratiques & erreurs fréquentes

- ☒ **Nom exact** : **view_item** (minuscule, underscore)
- ☒ **Minimum requis** dans **items[]** : **item_id** ou **item_name**
- ☒ **Une seule** balise GA4 Config
- ☒ **Ne pas envoyer **ecommerce.items** à la fois** via “Send Ecommerce data” **et** via Event Parameters (doublons)

- ❌ Ne pas déclencher sur "All Pages" (utilise **Custom Event = view_item**)

LES RÈGLES UTM TRACKING (WEB & ADS)

Les UTM sont des paramètres ajoutés à une URL pour tracker précisément l'origine et la performance des visites sur un site web dans des outils comme Google Analytics. Ils permettent à un expert en tracking web de mesurer l'efficacité des campagnes marketing, d'attribuer les conversions aux bonnes sources et d'optimiser le ROI en identifiant les canaux les plus performants.

Paramètres Principaux

Les cinq paramètres UTM standard sont :

- utm_source : Identifie la source (ex. : google, newsletter, facebook).
- utm_medium : Précise le canal (ex. : cpc, email...).
- utm_campaign : Nom de la campagne (ex. : promo_noel_2025) pour tracker des promotions spécifiques.
- utm_term : Mot-clé payant (ex. : chaussures running) pour les recherches sponsorisées.
- utm_content : Différencie les variantes (ex. : banniere_bleue vs banniere_rouge) pour les A/B tests.

Tous sont optionnels mais source, medium et campaign forment le trio minimal pour un tracking fiable.

Comment Créer des UTM

Utilisez l'URL de base, ajoutez "?" puis les paramètres séparés par "&", en encodant les espaces par "+" ou "%20" (ex. :

https://example.com/?utm_source=facebook&utm_medium=social&utm_campaign=launch

).

Des builders gratuits comme ceux de Google, HubSpot ou <https://ga-dev-tools.google/campaign-url-builder/> génèrent et valident ces liens automatiquement. Un expert encode toujours les URLs pour éviter les erreurs de parsing et utilise des conventions cohérentes (minuscules, tirets) pour une analyse propre.

À quoi servent les UTM concrètement ?

Les UTM servent à répondre à ces questions business 📌

- Cette vente vient de **Facebook Ads ou Instagram Ads** ?
- Quel **email** a généré le plus de conversions ?
- Quelle **annonce** fonctionne le mieux ?

- Quel **influenceur** / **lien bio** rapporte du CA ?
- Quel **CTA** dans une landing page est le plus performant ?

👉 **Les UTM = base de l'attribution marketing**

Structure d'une URL avec UTM

🔗 URL sans UTM

<https://www.monsite.com/produit>

🔗 URL avec UTM

[https://www.monsite.com/produit?](https://www.monsite.com/produit?utm_source=facebook)

[utm_source=facebook](https://www.monsite.com/produit?utm_source=facebook)

[&utm_medium=cpc](https://www.monsite.com/produit?utm_source=facebook)

[&utm_campaign=summer_sale](https://www.monsite.com/produit?utm_source=facebook)

[&utm_content=video_1](https://www.monsite.com/produit?utm_source=facebook)

[&utm_term=chaussure_running](https://www.monsite.com/produit?utm_source=facebook)

Les 5 paramètres UTM (à connaître par cœur)

✅ 1. utm_source (OBLIGATOIRE)

👉 D'où vient le trafic ?

Exemples :

- facebook
- instagram
- google
- newsletter
- tiktok
- linkedin
- whatsapp

Règle → Toujours la **plateforme / source**

✅ 2. utm_medium (OBLIGATOIRE)

👉 Quel type de canal marketing ?

Exemples standards :

- cpc / paid_search (paid ads)
- paid_social
- organic_social
- email
- organic_search
- referral
- affiliate
- direct



Règle clé



Le **medium doit rester cohérent dans tout l'écosystème**



Mauvais :

- ads / paid / boost / sponsorisé (mélange)



Bon :

- cpc ou paid_social (choisir 1 et garder partout)



3. utm_campaign (OBLIGATOIRE)



Nom de la campagne marketing

Exemples :

- soldes_ete_2025
- ramadan_offer
- lancement_app
- black_friday



Règles

- Pas d'espace
- Pas d'accent
- Pas de caractères spéciaux
- Snake_case recommandé



4. utm_content (OPTIONNEL mais TRÈS IMPORTANT)



Différencier les créas / CTA / formats

Exemples :

- video_15s
- carousel_1

- banner_homepage
- cta_buy_now
- story_instagram

À utiliser pour

- A/B testing
- Comparer visuels
- Comparer messages

5. utm_term (OPTIONNEL)

Mot-clé (surtout Search)

Exemples :

- chaussure_running
- parfum_femme
- salle_sport_tunis

Règles STRICTES à respecter :

1. Toujours en minuscules

 Facebook / CPC

 facebook / cpc

 Les outils d'analytics sont case-sensitive

2. Ne jamais changer la logique source / medium

 facebook → fb → meta

 paid → ads → sponsorisé

 Choisir UNE convention et la garder

CRÉER UN CANAL PERSONNALISÉ DANS GA4

(Cas : **utm_medium=influencer**)

GA4 fonctionne en 2 niveaux :

 Niveau 1 – Les UTM (données brutes)

Les UTM (**utm_source**, **utm_medium**, etc.) sont :

- Envoyées via l'URL
- Stockées telles quelles dans GA4
- Jamais modifiées par GA4

Exemple : `utm_medium=influencer`

GA4 reçoit bien la valeur influencer.

◆ Niveau 2 – Channel Grouping (classification)

GA4 applique ensuite des règles automatiques pour classer les sessions dans des canaux :

- Paid Social
- Organic Social
- Email
- Paid Search
- Referral
- etc.

👉 Si le medium n'est pas reconnu comme **influenceur**, GA4 classe la session dans : ❌ Unassigned

Solution officielle GA4 : créer un Channel Grouping personnalisé

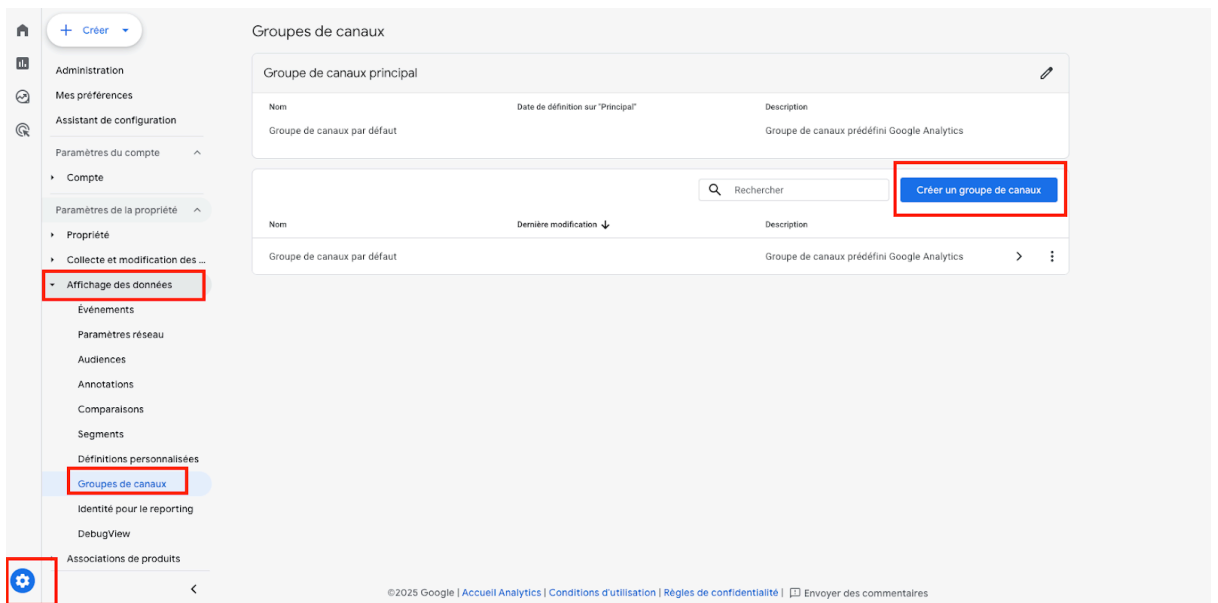
✅ Principe

Si on invente un `utm_medium` → on doit créer la règle qui va avec dans GA4.

Il ne s'agit PAS :

- d'un paramètre UTM
- ni d'un réglage de campagne
- ni d'un événement

👉 Il s'agit d'un Channel Grouping.



🔧 Étape 1 – Créer un nouveau Channel Group

- Nom : **Custom Channel Group – Marketing**

👉 Bonne pratique : ne pas modifier l'original

🔧 Étape 2 – Ajouter un nouveau canal

- Channel name : **Influencer**

🔧 Étape 3 – Définir la règle du canal

Règle recommandée (simple et propre)

- Dimension : Session medium
- Condition : equals
- Valeur : influencer

Toute session dont **utm_medium=influencer** → sera classée dans le canal Influencer

Les paramètres d'attribution facebook

7 jours clic / 1 jour vue

1 Définition (à retenir absolument)

La **fenêtre d'attribution** définit **pendant combien de temps une conversion peut être attribuée à une action marketing** (clic ou affichage).

👉 **7 jours clic / 1 jour vue** signifie :

- **Clic :**
Si un utilisateur **clique sur une publicité** et réalise une conversion **dans les 7 jours**,
👉 la conversion est attribuée à la campagne.
- **Vue (impression) :**
Si un utilisateur **voit une publicité sans cliquer** et convertit **dans les 24 heures**,
👉 la conversion est aussi attribuée à la campagne.

2 Pourquoi cette règle existe ?

Dans le parcours client réel :

- L'utilisateur **ne convertit pas toujours immédiatement**
- Il peut :
 - voir une pub → réfléchir
 - cliquer plus tard
 - acheter plusieurs jours après

👉 Cette fenêtre permet donc de **mieux refléter l'influence réelle du marketing**, sans être trop large (et fausser les résultats).

3 Exemple simple (niveau débutant)

🧠 Cas n°1 – Attribution par clic

- Jour 1 : un utilisateur **clique** sur une publicité Facebook
- Jour 5 : il achète sur le site

✅ Conversion attribuée à la publicité
(car l'achat a eu lieu **dans les 7 jours après le clic**)

👁️ Cas n°2 – Attribution par vue

- Jour 1 : un utilisateur **voit** une publicité (sans cliquer)
- Jour 2 (12h plus tard) : il tape le nom du site sur Google et achète

✅ Conversion attribuée à la publicité
(car l'achat a eu lieu **dans les 24h après la vue**)

❌ Cas n°3 – Pas d'attribution

- Jour 1 : vue de la publicité
- Jour 3 : achat

❌ Conversion **non attribuée**
(car la conversion a eu lieu **au-delà du 1 jour vue**)

4 Pourquoi **7 jours clic / 1 jour vue** est un standard ?

Cette configuration est aujourd'hui considérée comme **un bon équilibre** :

Trop court ❌	Équilibré ✅	Trop long ❌
Sous-estime la performance	Reflète le vrai comportement	Sur-attribution
Décisions biaisées	Décisions data-driven	ROI artificiellement gonflé

👉 C'est **le standard recommandé** sur la plupart des plateformes publicitaires modernes.

Conclusion : La fenêtre d'attribution **7 jours clic / 1 jour** vue permet de relier intelligemment les actions marketing aux conversions, en tenant compte du délai naturel de décision du consommateur.

Elle constitue une base fiable pour analyser la performance, comparer les campagnes et prendre des décisions data-driven, tout en gardant un esprit critique sur les résultats observés.