



*La logique évolutive de BTP n'est pas technique avant tout. C'est une réponse architecturale à l'évolution des ERP, des métiers et du cloud.*

# Le point de départ : pourquoi SAP a dû changer de logique

Ancien monde (ECC / ERP classique)

L'ERP était :

- le **lieu du process**
- le **lieu du code**
- le **lieu de l'intégration**
- le **lieu de l'innovation**

Résultat :

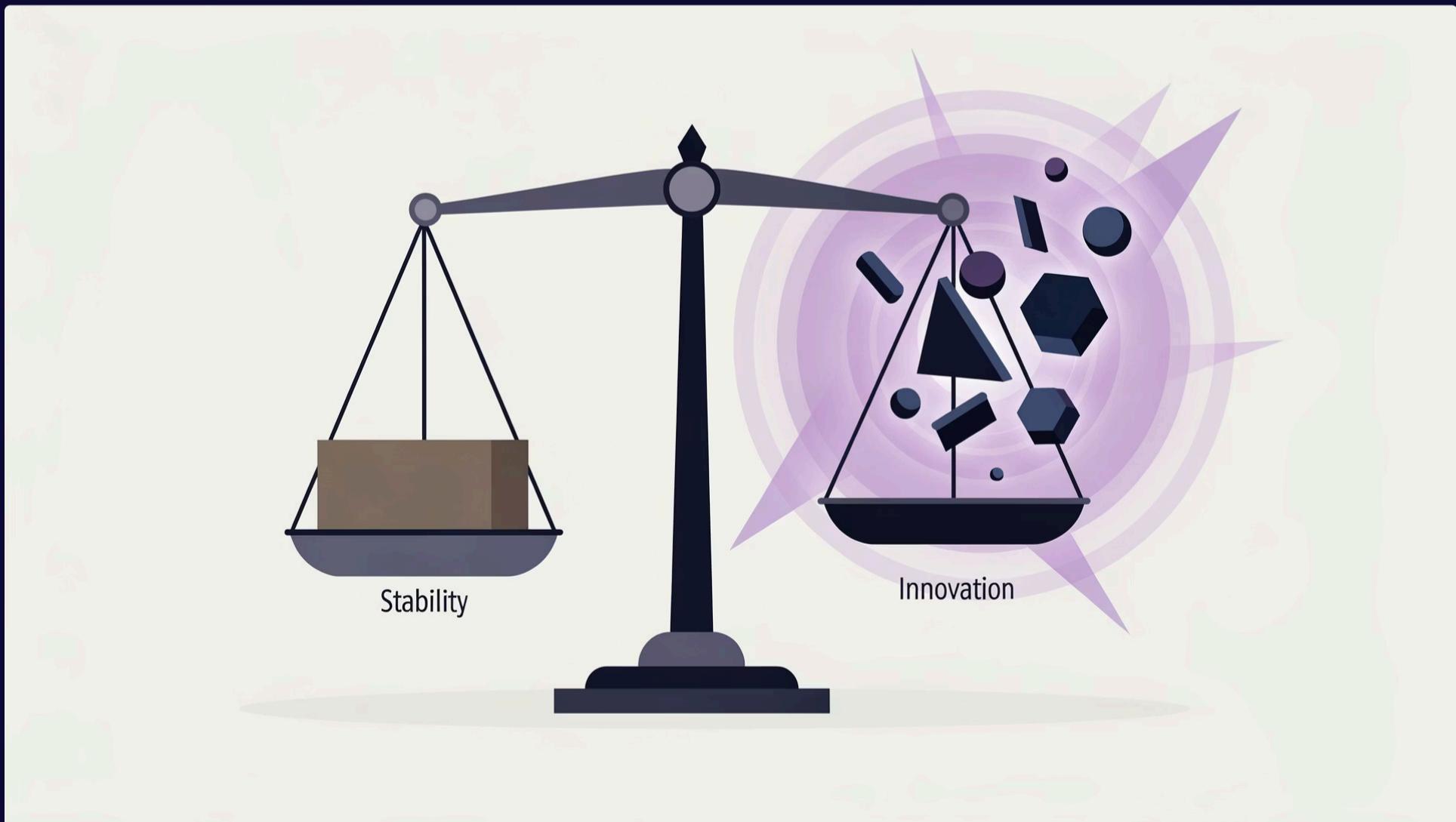
- Couplage fort
- Modifications Z
- Régressions à chaque upgrade
- ERP = **système rigide**



# Problème clé identifié par SAP

**Un ERP ne peut plus être à la fois stable  
ET innovant**

SAP a donc séparé les responsabilités.

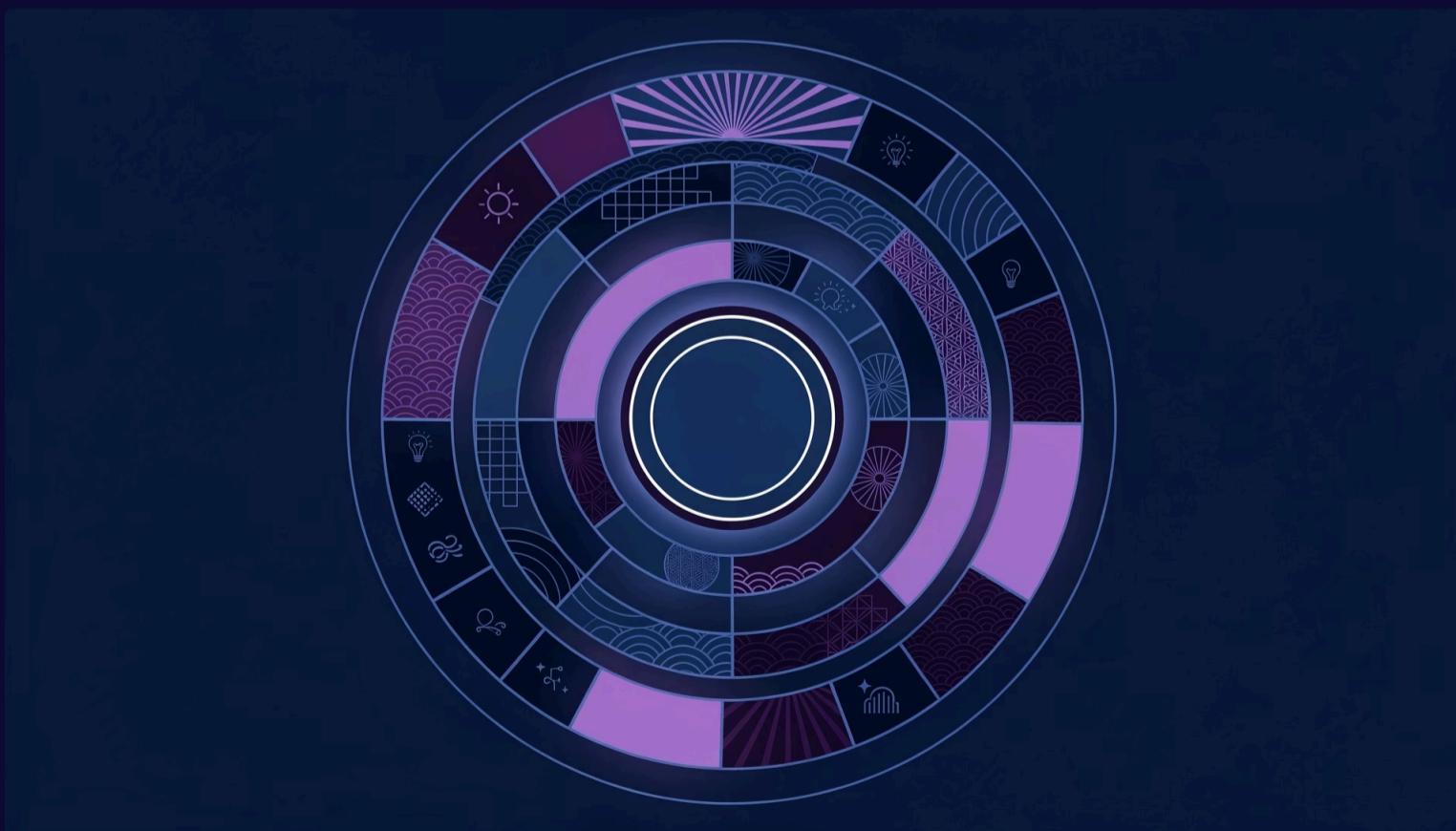


# Le principe fondateur de la logique évolutive BTP

Séparer stabilité et changement

Dimension	Rôle
S/4HANA	Exécution des processus cœur
SAP BTP	Évolution, extension, innovation

C'est le socle de toute la logique BTP



# Clean Core : le socle de la logique évolutive

**Clean Core, ce n'est PAS :**

- "Pas de développement"
- "Tout en standard"
- "On ne touche plus à rien"

**Clean Core, c'est :**

- **Zéro modification du standard**
- **Extensions contrôlées**
- **Découplage technique**

**BTP est l'outil opérationnel du Clean Core**

Règle mentale de consultant :

Tout ce qui change souvent ne doit pas être dans S/4

# La logique évolutive en couches

SAP BTP n'évolue pas "d'un coup".

Il suit une progression logique, presque naturelle.



# Couche 1

## Accéder sans dépendre (CONNECT)

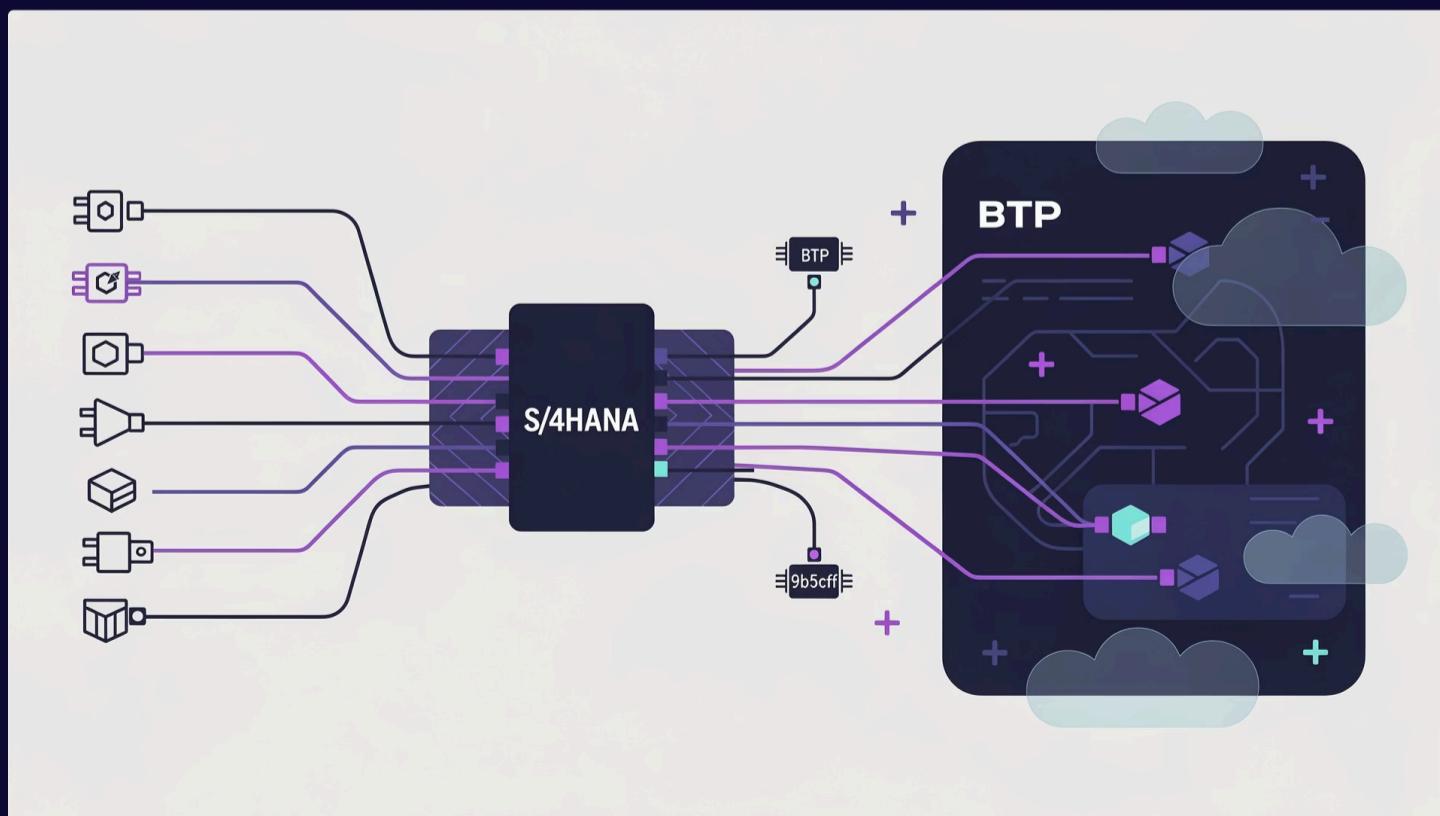
**Problème à résoudre :**

Comment utiliser les données S/4 sans casser le core ?

**Réponse BTP :**

- APIs standard
- OData / REST
- Events
- Cloud Connector

**Accès ≠ dépendance**



# Couche 2

## Étendre sans modifier (EXTEND)

**Problème :**  
**Où mettre la logique métier spécifique ?**

**Réponse BTP :**

- Side-by-side extensibility
- ABAP Cloud
- CAP
- SAP Build Apps

La logique évolutive dit :

*La logique métier volatile vit hors de l'ERP*

**Exemple typique :**

- Calcul spécifique
- Workflow client
- Validation conditionnelle
- Application métier satellite

# Couche 3

## Intégrer sans rigidité (INTEGRATE)

Problème :

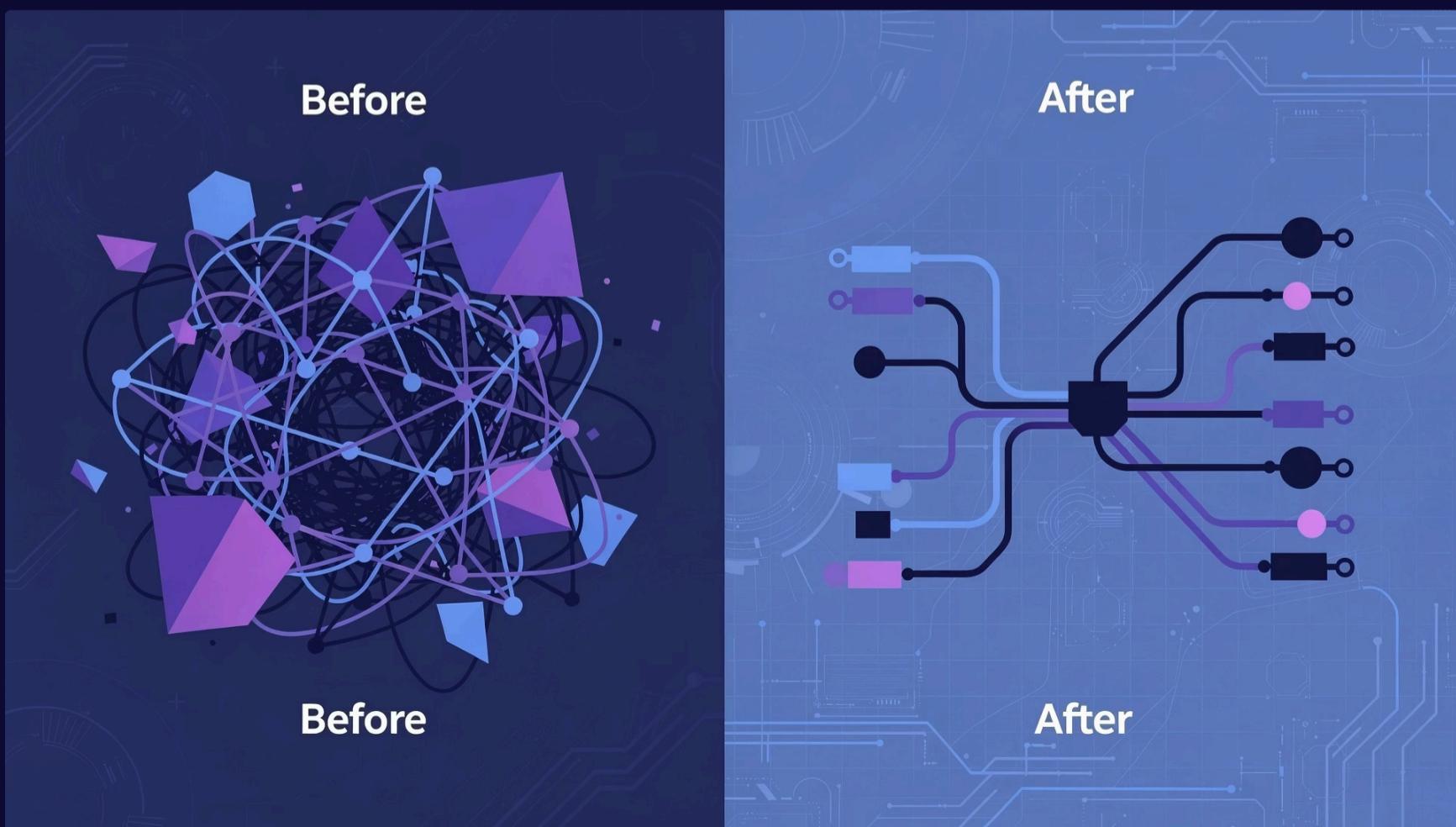
Comment faire évoluer le SI sans effet domino ?

Réponse BTP :

- SAP Integration Suite
- iFlows
- Event-driven architecture
- API Management

On passe de :

intégration point-à-point à **orchestration évolutive**



# Couche 4

## Automatiser sans complexifier (AUTOMATE)

Problème :

Comment adapter les processus rapidement ?

Réponse BTP :

- SAP Build Process Automation
- Business Rules
- Workflows low-code

Les métiers peuvent évoluer **sans toucher au core ERP**

Logique clé :

Le process devient **configurable**, pas codé

# Couche 5

## Innover sans risque (INNOVATE)

Problème :

Où tester l'avenir sans mettre en danger l'existant ?

Réponse BTP :

- AI / Joule
- Event Mesh
- GenAI
- HANA Cloud
- Extensions expérimentales

BTP devient un **laboratoire contrôlé**



# La logique évolutive dans le temps

SAP BTP permet une évolution **progressive** :

01

---

**D'abord connecter**

02

---

**Puis étendre**

03

---

**Puis intégrer**

04

---

**Puis automatiser**

05

---

**Puis innover**

**Pas besoin de tout faire dès le départ**

Bonne pratique de consultant :

Toujours commencer par le minimum d'évolution utile

# Ce que change vraiment BTP dans la posture IT

## Avant

- IT = exécutant
- ERP = contrainte
- Projets longs

## Avec BTP

- IT = architecte de l'évolution
- ERP = fondation stable
- Innovation continue

SAP BTP transforme :

**l'ERP en plateforme**, et l'IT en moteur d'évolution

# Message clé à retenir

**SAP BTP n'est pas une plateforme technique.  
C'est une logique d'évolution maîtrisée du SI SAP.**

**S/4 = stabilité**

**BTP = mouvement**

**Clean Core = règle du jeu**