Fiche mémo | Leçon 15 | 27 min

# Boucles, itérateurs et agrégateurs pour traiter une liste ligne par ligne

Quand un flux reçoit une liste, il la découpe pour traiter chaque élément, puis la regroupe pour en rendre compte. n8n traite déjà tous les éléments sans boucle ; Make découpe avec l'itérateur et regroupe avec un agrégateur ; Zapier passe par Looping. Chaque passage peut se payer : filtrez avant de boucler et testez sur trois lignes.

## Objectifs

- Repérer quand un flux reçoit une liste, et choisir entre découper, traiter tel quel ou regrouper
- Construire un traitement ligne par ligne suivi d'un récapitulatif unique dans n8n, Make ou Zapier
- Chiffrer le coût d'une boucle en exécutions, crédits ou tâches, et la protéger contre l'emballement

## À retenir

- Un flux qui reçoit une liste la découpe pour agir sur chaque élément, puis la regroupe pour rendre compte en un seul message.
- Dans n8n, chaque nœud traite déjà tous les éléments reçus : Loop Over Items ne sert qu'aux exceptions, aux paquets avec pause et à la pagination.
- Dans Make, l'itérateur découpe une liste contenue dans un lot et l'agrégateur regroupe à partir de son module source ; chaque lot traité se paie en crédits.
- Dans Zapier, chaque action après Looping s'exécute à chaque tour ; un filtre sur le dernier tour réserve le récapitulatif à la fin.
- Filtrez avant de découper et limitez le premier essai à trois lignes : une boucle emballée coûte des milliers d'actions.

## La demande à recopier

**Demande à recopier** **Faire écrire le modèle de relance qui servira à chaque brouillon**

```
Rôle : tu es l'assistante commerciale d'une menuiserie de 12 personnes à Pessac.
Contexte : ce texte sera le corps d'un brouillon de relance créé automatiquement pour chaque
devis sans réponse depuis une semaine. Les champs disponibles sont {{ $json['Prénom'] }}, {{
$json['Nom'] }}, {{ $json['ID demande'] }}, {{ $json['Type de projet'] }} et {{
$json['Responsable'] }}.
Tâche : écris le modèle de relance, en gardant ces champs tels quels là où ils doivent
apparaître.
Format : objet, puis six lignes au plus, vouvoiement, une seule question à la fin. Aucun prix,
aucune remise, aucun délai : ils ne sont pas dans les champs.
```

**Note :** Relisez le modèle une fois avec de vraies valeurs : une expression mal recopiée laisse « {{ » dans le courriel du client.

## Le code à recopier

**Le Filter « À relancer aujourd'hui »** JAVASCRIPT

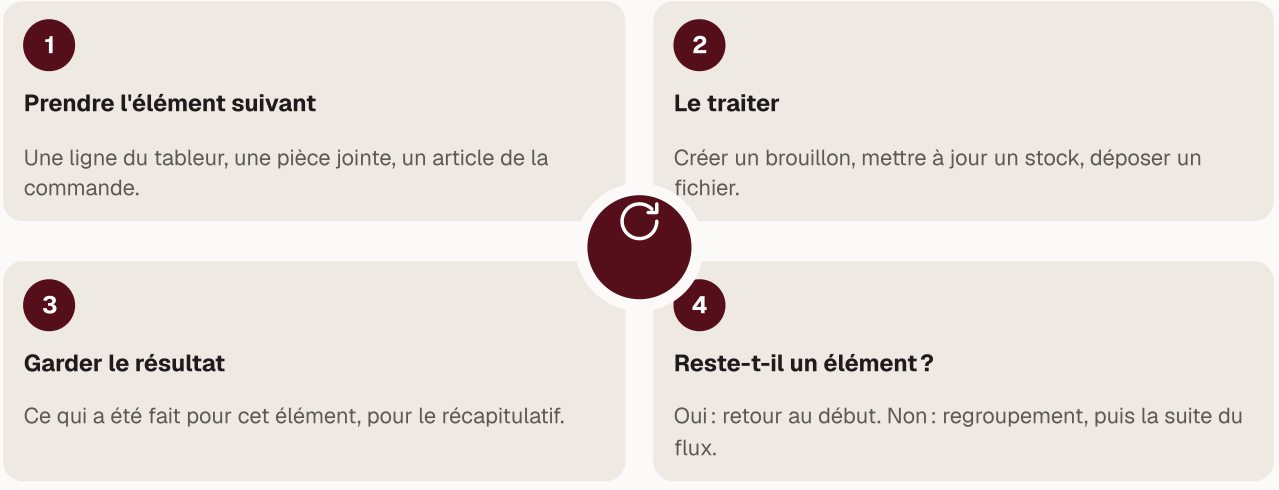
```
{{ DateTime.fromFormat($json['Relance prévue le'], 'dd/MM/yyyy') <= $today }}
```

Si une cellule est vide ou mal écrite, la date est invalide et la comparaison renvoie false : la ligne ne passe pas. C'est voulu, et la leçon « Dates, montants, noms et adresses, remettre les données en forme en chemin » montre comment éviter ces cellules.

## Les erreurs fréquentes

| L'erreur, et pourquoi elle coûte | La correction |
|---|---|
| **Ajouter un nœud Loop Over Items devant chaque action de n8n, par habitude venue d'un autre outil.** Chaque nœud de n8n traite déjà tous les éléments reçus : la boucle ajoutée complique le flux, rend l'historique plus difficile à lire et crée des occasions de boucle sans fin. | Laissez les nœuds traiter la liste. Réservez Loop Over Items aux nœuds qui ne lisent que le premier élément, aux API qui demandent des paquets avec une pause et à la pagination. |
| **Régler l'agrégateur de Make sur le mauvais module source, souvent le déclencheur.** Le regroupement ne commence pas là où la liste a été découpée : le récapitulatif est incomplet, ou un récapitulatif part par lot. | Réglez le « Source Module » sur l'itérateur ou sur la recherche qui a produit les lots, et vérifiez sur le canevas la zone grisée entre les deux modules. |
| **Lancer la première exécution sur le tableur entier, sans filtre ni limite.** Une boucle traite tout ce qu'elle reçoit : des milliers d'actions facturées chez Make ou Zapier, et des courriels envoyés à des clients qui n'étaient pas concernés. | Filtrez avant de découper, ajoutez un nœud Limit ou le champ « Limit » de la recherche à 3 pour le premier essai, et travaillez sur une copie du tableur. |
| **Envoyer une notification par ligne traitée au responsable.** Douze, vingt, quarante messages le même matin : il cesse de les lire, et l'alerte qui compte se perd au milieu des autres. | Regroupez avec Aggregate dans n8n, un agrégateur dans Make ou un filtre sur le dernier tour dans Zapier, et envoyez un seul récapitulatif. |

### Ce que fait une boucle




Le même traitement, élément après élément, jusqu'au dernier. Puis le regroupement.