

Faut-il automatiser cette tâche : l'arbre de décision

Cinq questions dans l'ordre pour savoir si une tâche mérite d'être automatisée, quatre sorties qui disent non, le calcul de rentabilité au coût horaire réel et le test des six mois.

BLYX-AGENCY.COM · 9 SEPTEMBRE 2026 · PAR MELVIN BRETON

La plupart des demandes d'automatisation arrivent sous la forme d'une liste. Trois ou quatre tâches qui agacent, avec la question qui va avec : combien ça coûte de les automatiser. C'est la deuxième question. La première, personne ne se la pose, parce qu'elle est désagréable : est-ce que cette tâche mérite qu'on dépense quoi que ce soit dessus.

La réponse courte. Une tâche répétitive n'est pas forcément une tâche à automatiser. Avant de chiffrer quoi que ce soit, cinq questions se posent dans un ordre précis, et elles ont six sorties possibles. Une seule dit d'automatiser. Les cinq autres coûtent moins cher, vont plus vite, et règlent le problème dans la majorité des cas.

Tu y trouveras l'arbre complet avec ses cinq questions, le détail de chaque sortie et ce qu'elle implique concrètement, le calcul de rentabilité fait avec le coût horaire réel d'un salarié en France, trois tâches courantes passées dans l'arbre du début à la fin, les cinq erreurs qui font sauter le raisonnement, et le test des six mois à faire avant de signer un devis.



1. Pourquoi « comment automatiser » arrive

trop tôt



Une automatisation ne rend pas une tâche meilleure. Elle la rend plus rapide et plus difficile à remettre en cause. C'est une différence de nature, pas de degré.

Tant qu'une tâche est faite à la main, elle reste discutable. Quelqu'un peut la trouver absurde et arrêter de la faire. Le jour où elle tourne toute seule, plus personne ne la voit, plus personne ne la remet en question, et elle survit à la raison qui l'a fait naître. Le coût ne disparaît pas, il devient invisible.

C'est pour ça que l'ordre des questions compte autant que les questions elles-mêmes. Chercher l'outil avant d'avoir jugé la tâche revient à choisir un moyen de transport avant de savoir où l'on va.

Le seul projet d'automatisation qui ne déçoit jamais est celui qu'on a décidé de ne pas faire. Il ne coûte rien, il ne casse pas, il ne demande aucune maintenance, et le temps qu'il libère est immédiat.

Les chiffres publics vont dans le même sens. Dans son rapport de 2024 sur les causes racines de l'échec des projets d'intelligence artificielle, la [RAND Corporation](#) rappelle que plus de 80 % d'entre eux échouent, soit deux fois le taux d'échec des projets informatiques qui ne touchent pas à l'IA. Gartner, de son côté, [prévoit que plus de 40 % des projets d'IA agentique seront annulés d'ici fin 2027](#), et cite trois causes : des coûts qui dérapent, une valeur métier floue, des contrôles de risque insuffisants. Ces trois causes se décident en amont, avant



première ligne de code. C'est exactement ce que l'arbre ci-dessous

 Blyx. Sources relevées le 9 septembre 2026.



Une précision de périmètre avant d'entrer dedans. Cet arbre qualifie **une tâche prise isolément** et répond non dans la majorité des cas. Si tu as déjà une liste de tâches jugées automatisables et que tu cherches par laquelle commencer, c'est un autre travail, traité dans [quelles tâches automatiser en premier](#). Et si la décision d'y aller est déjà prise, la question devient qui fait quoi entre l'humain et la machine, ce que détaille [la méthode humain plus IA](#).

2. L'arbre en entier

Cinq questions, posées dans cet ordre. Tu descends tant que la réponse est oui. La première réponse négative te donne ta sortie, et tu t'arrêtes là.

#	La question	Si la réponse est non	Ce que ça coûte
1	Si tu arrêtais cette tâche demain, quelqu'un s'en apercevrait ?	Supprimer	Zéro euro
2	Quelqu'un d'autre que toi sait l'expliquer pas à pas ?	Écrire le process d'abord	Quelques heures
3	Deux personnes qui la font sortent le même résultat ?	Laisser humain, et l'outiller	Faible à moyen
4	Ça prend plus d'une heure par semaine ?	Simplifier, sans rien développer	Zéro à faible





La question

Blyx

Si la réponse est
non

Ce que ça coûte



5

Les données d'entrée sont
propres et récupérables ?

**Nettoyer la source
d'abord**

Variable

Cinq fois oui

Automatiser

Quelques
milliers d'euros
et plus

Comment lire ce tableau. L'ordre n'est pas décoratif. Il va du moins cher au plus cher, et de la question la plus dérangeante à la plus technique. Une tâche qui sort à la question 1 n'a jamais besoin qu'on réponde aux quatre suivantes. C'est ce qui rend l'exercice rapide : la plupart des tâches sortent avant la fin, en quelques minutes.

Une seule règle d'usage, et elle est stricte : l'arbre s'applique à **une tâche précise**, pas à un service ni à un métier. « La saisie des factures fournisseurs », oui. « La comptabilité », non. Une tâche qu'on n'arrive pas à nommer en une phrase courte est en réalité un ensemble de tâches, et chacune mérite sa propre descente.

3. Question 1 : si tu arrêtais demain, qui s'en apercevrait ?

C'est la question que personne ne pose, et c'est celle qui élimine le plus de candidats.

Beaucoup de tâches survivent à leur raison d'être. Un tableau de suivi créé pour un client qui est parti il y a deux ans. Un rapport hebdomadaire lancé pour rassurer un banquier lors d'une demande de prêt. Un export mensuel que plus personne n'ouvre mais que tout le monde



Le test concret. Arrête la tâche. Ne préviens personne. Attends. Si au bout de trois ou quatre cycles complets personne n'est venu la réclamer, tu as ta réponse, et elle est gratuite. Si quelqu'un vient, tu as appris quelque chose de plus précieux encore : tu sais désormais qui utilise le résultat et pour quoi faire, ce qui changera la façon de l'automatiser plus tard.

Trois signaux qui trahissent une tâche fantôme, sans même faire le test :

- ✓ Personne ne sait dire qui lit le résultat, ni ce qu'il en fait
- ✓ Le format n'a pas changé depuis des années, alors que l'activité, elle, a changé
- ✓ La tâche produit un document que son destinataire retraite lui-même à sa réception

***Attention au réflexe inverse.** Une tâche invisible n'est pas forcément inutile. Une sauvegarde dont personne ne parle jamais est exactement dans ce cas : personne ne s'apercevrait de son arrêt, jusqu'au jour où tout le monde s'en apercevrait d'un coup. Pose-toi la question du sinistre, pas seulement celle de la plainte.*

Avant de décider, chiffre ce que la tâche te coûte réellement aujourd'hui. La méthode complète est dans [comment calculer le retour sur investissement d'un projet IA](#), et elle sert autant à justifier une automatisation qu'à justifier une suppression.



4. Question 2 : est-ce que quelqu'un sait

l'expliquer sans toi ?



Prends la personne qui fait la tâche. Demande-lui de l'écrire pas à pas, sans rien omettre, pour quelqu'un qui arriverait demain. Si elle n'y arrive pas, ou si le document obtenu ne permet pas à un tiers de refaire le travail, la tâche n'est pas automatisable aujourd'hui.

Ce n'est pas une question de discipline documentaire, c'est une question mécanique. Une automatisation est un mode opératoire exécuté par une machine. S'il n'existe aucun mode opératoire, il faudra bien que quelqu'un l'invente au moment de développer, et ce quelqu'un ne connaît pas ton métier. Le résultat sera une version approximative du travail réel, avec des écarts qui se découvriront en production, un par un, chacun au pire moment.

Ce que tu as	Ce que ça permet	Ce qu'il manque
Rien d'écrit, tout dans une tête	Rien, et le poste est un risque	Le mode opératoire
Une liste d'étapes générale	Une discussion, pas un cahier des charges	Les cas particuliers et les exceptions
Le pas à pas avec les exceptions et les chiffres	Un chiffrage sérieux et un projet tenable	Rien, tu peux avancer

Ce que cette sortie rapporte, même sans automatisation derrière.

Écrire le mode opératoire d'un poste, c'est transformer un savoir qui tient dans une tête en un actif qui appartient à l'entreprise. Ça sécurise un départ, ça raccourcit une formation, ça rend une délégation possible. Beaucoup de dirigeants découvrent à ce moment que la tâche qu'ils voulaient automatiser était surtout une tâche qu'ils



voulaient arrêter de faire eux-mêmes, ce qui n'est pas le même

 **Blyx** ne se règle pas avec le même budget.



C'est aussi la question éliminatoire côté technique : tant qu'un process n'est pas écrit, le débat sur l'outil à utiliser n'a pas de sens, comme le détaille [no-code ou full-code](#).

5. Question 3 : deux personnes obtiennent-elles le même résultat ?

Fais l'expérience mentalement. Deux personnes compétentes, le même dossier d'entrée, le même mode opératoire. Est-ce qu'elles produisent la même sortie ?

Si oui, la tâche est déterministe, et une machine peut la reprendre. Si non, quelque chose d'autre entre en jeu : le contexte du client, l'historique de la relation, une part de responsabilité, un arbitrage qui n'est écrit nulle part parce qu'il dépend de trop de choses.

Les écarts entre deux personnes ne sont pas un défaut à corriger, ils sont l'information. Ils te disent où se trouve la valeur ajoutée humaine dans cette tâche. C'est souvent une petite partie du travail total, et c'est justement celle qu'il ne faut pas confier à une machine.

Ce que veut dire outiller sans remplacer. La sortie n'est pas « ne rien faire ». Elle consiste à retirer la charge autour de la décision, en laissant la décision à la personne :

- ✓ Rassembler et présenter les éléments dont elle a besoin, au bon endroit et au bon moment



- ✓ Reprendre la mise en forme, l'envoi, l'archivage, une fois qu'elle a tranché

La frontière est simple à tenir : la machine prépare et exécute, la personne décide et signe. Le découpage précis entre ce qui revient à l'un et à l'autre est développé dans [la méthode humain plus IA](#), qui prend le relais dès que cette sortie est atteinte.

6. Question 4 : est-ce que ça prend plus d'une heure par semaine ?

C'est le seul endroit de l'arbre où un chiffre tranche. Il faut donc que le chiffre soit juste.

Une heure par semaine représente environ 45 heures par an, une fois retirés les congés et les jours fériés. Reste à savoir combien vaut une heure. Le coût horaire de la main d'oeuvre en France s'élevait à 43,70 euros en 2024 selon les données Eurostat reprises par l'[Insee](#) (relevé le 9 septembre 2026), et il approche les 45 euros début 2026. Ce chiffre couvre les entreprises de dix salariés et plus, dans l'industrie, la construction et les services, et il inclut les cotisations patronales.

Autrement dit, une heure par semaine coûte environ 2 000 euros par an à ton entreprise. Une automatisation simple se compte en milliers d'euros. La conclusion est arithmétique : sous une heure par semaine, le projet ne se rembourse pas dans l'année, et souvent pas dans les deux.



30 minutes par semaine	22 h	environ 970 €	Simplifier, ne rien développer
1 heure par semaine	45 h	environ 2 000 €	Seuil bas, seulement si la solution est simple
3 heures par semaine	135 h	environ 5 900 €	Le projet se rembourse dans l'année
1 heure par jour	225 h	environ 9 900 €	Priorité haute, à chiffrer tout de suite

Comment lire ce tableau. Les montants sont calculés à 44 euros de l'heure, le repère public. Si tes salaires sont plus bas, refais le calcul avec ton propre chiffre : ce qui compte, c'est la méthode, pas mon multiplicateur. Et compte le temps réel, pas le temps ressenti. Une tâche pénible paraît toujours plus longue qu'elle ne l'est, une tâche fragmentée en dix passages de trois minutes paraît toujours plus courte que son total.

La sortie « simplifier » est la plus sous-estimée de l'arbre. Sous le seuil, il reste presque toujours du temps à récupérer sans écrire une ligne de code : supprimer un champ que personne ne remplit, retirer une validation qui n'a jamais rien bloqué, arrêter de saisir deux fois la même information à deux endroits, remplacer un aller-retour par mail par un formulaire. C'est gratuit, c'est immédiat, et ça marche aussi au dessus du seuil, où ça réduit le périmètre à automatiser avant de le chiffrer.

Attention au piège de périmètre à ce stade. Une tâche est rarement uniforme : elle contient des blocs qui se répètent à l'identique et de



blocs qui varient à chaque fois. Ce sont les premiers qu'on automatise,

 **Blyx** entière. La démonstration déroulée sur un cas complet est dans [supprimer cinq heures par semaine sur un process critique](#).

7. Question 5 : les données d'entrée sont-elles propres et récupérables ?

Dernière question avant le feu vert, et celle qui fait dérailler le plus de projets une fois lancés. Une automatisation ne crée pas d'information, elle en déplace. Si ce qui entre est incohérent, ce qui sort le sera aussi, plus vite et en plus grand nombre.

Trois conditions, cumulatives :

- 1. Un endroit unique.** L'information vit à une adresse connue, pas dans trois fichiers, deux boîtes mail et une tête.
- 2. Un format stable.** Les mêmes colonnes, les mêmes intitulés, les mêmes unités d'une fois sur l'autre. Un fichier dont la structure change chaque mois n'est pas une source, c'est une surprise mensuelle.
- 3. Une sortie possible.** Le logiciel qui détient la donnée sait l'exporter, ou expose une interface pour aller la chercher. Un écran qu'on ne peut que lire n'est pas une source de données.

Si l'une des trois manque, la sortie est de traiter la source avant tout le reste. Le sujet dépasse largement cet arbre et il est traité en entier dans [des données propres avant l'IA](#), qui explique surtout jusqu'où aller : le nettoyage total est un chantier qui meurt avant d'avoir livré, l'objectif est un niveau de qualité suffisant pour l'usage visé, pas la perfection.



8. Cinq fois oui : automatiser, et à quelles



Une tâche qui a passé les cinq questions est utile, écrite, déterministe, coûteuse et alimentée proprement. C'est un bon projet. Il reste une dernière vérification, qui ne porte plus sur la tâche mais sur ce qui se passe après la livraison.

Appelons ça le test des six mois. Quatre questions, quatre réponses nominatives. Si l'une d'elles n'a pas de nom en face, le projet n'est pas prêt à être signé.

La question	Ce qu'une bonne réponse contient
Qui contrôle le résultat ?	Un nom, une fréquence, et le moyen concret de repérer une erreur
Comment une erreur se voit ?	Une alerte, un écart visible, un contrôle de cohérence. Pas « on s'en apercevra »
Qui répare, et sous quel délai ?	Un nom, un contrat de garantie, un délai écrit
Qui possède le code et les accès ?	Toi. Sinon tu as changé de dépendance, pas gagné en autonomie

Une automatisation n'est pas un achat, c'est une pièce d'équipement. Elle casse quand un outil d'en face change son interface, quand ton process évolue, quand la personne qui l'a montée s'en va. Le budget honnête compte la construction et l'entretien, pas seulement la construction.



C'est seulement ici, une fois que tout ce qui précède est réglé, que la question de l'outil se pose : automatisation no-code, développement sur mesure, ou combinaison des deux. Les critères de ce choix sont détaillés dans [no-code ou full-code](#), et le premier branchement concret dans [intégrer l'IA dans une PME](#).

9. Trois tâches passées dans l'arbre

Les exemples qui suivent sont des illustrations construites pour montrer la mécanique, pas des cas clients.

9.1 La relance des factures impayées

Question 1 : si personne ne relance, l'argent ne rentre pas. Quelqu'un s'en aperçoit, et vite. On continue. Question 2 : le process est écrit, ou s'écrit en une heure. Relance à sept jours, deuxième à quinze, appel à trente. On continue. Question 3 : deux personnes obtiennent le même résultat, parce que les règles sont des dates et des montants. On continue. Question 4 : deux heures par semaine, soit près de 4 000 euros par an. On continue. Question 5 : les données vivent dans l'outil de facturation, qui sait les exporter. On continue.

Verdict : automatiser, avec une réserve sur le test des six mois. La relance part au nom de l'entreprise et touche des clients : le contrôle doit être quotidien les premières semaines, et un client en litige ne doit jamais recevoir de relance automatique.

9.2 Le rapport d'activité hebdomadaire

Question 1 : arrêté pendant un mois, personne ne l'a réclamé. Sorti immédiate.



Verdict : supprimer. Le projet le plus rentable de la liste, et il ne coûte

rien. Le projet libéré est disponible la semaine suivante, pas dans six mois.  

9.3 La réponse aux demandes de devis entrantes

Question 1 : évidemment oui, c'est du chiffre d'affaires. On continue.

Question 2 : le process existe, il est écrit. On continue. Question 3 : non.

Deux commerciaux ne répondent pas la même chose à la même demande, parce que le prix dépend du client, de la charge de l'atelier et de l'envie de prendre le dossier. Sortie.

Verdict : laisser humain, et outiller. La machine rassemble l'historique du client, préremplit la trame et calcule les montants ; la personne arbitre le prix et signe. Le gain est réel, la décision reste là où elle doit être.

10. Les cinq erreurs qui font sauter l'arbre

- 1. Sauter la question 1 parce qu'elle est gênante.** Supprimer une tâche revient souvent à admettre qu'on la faisait pour rien depuis des années. C'est inconfortable, et c'est le gain le plus rapide disponible.
- 2. Prendre le temps ressenti pour le temps réel.** Sans mesure, on surestime ce qui agace et on sous-estime ce qui se fragmente. Note les durées pendant deux semaines avant de trancher à la question 4.
- 3. Automatiser la tâche entière au lieu du bloc qui se répète.** C'est ce qui transforme un projet de deux semaines en projet de deux mois, pour un gain identique.



4. Oublier la maintenance dans le calcul. Une automatisation qui



une demi-journée de reprise par trimestre consomme une part significative du temps qu'elle libère.

5. Automatiser ce que personne ne saura vérifier. Une erreur silencieuse qui tourne pendant six mois coûte plus cher que la tâche manuelle qu'elle a remplacée.

11. Comment s'en servir en réunion

L'arbre est fait pour être déroulé à voix haute, avec les personnes qui font le travail, pas dans un bureau à côté.

Le déroulé qui fonctionne tient en quatre étapes. Lister les tâches candidates, une ligne chacune, formulée assez précisément pour qu'on sache de quoi on parle. Descendre l'arbre pour chacune, à haute voix, en notant la question de sortie. Regrouper ensuite par sortie plutôt que par tâche, ce qui donne quatre chantiers distincts : ce qu'on supprime cette semaine, ce qu'on écrit ce mois-ci, ce qu'on simplifie sans budget, ce qu'on chiffre. Ne chiffrer qu'en dernier, sur ce qui reste.

Compte quelques minutes par tâche. La plupart sortent à la première ou à la deuxième question, et c'est le but : l'arbre est fait pour éliminer vite, pas pour justifier un projet.

Ce travail de décision, on peut aussi le mener avec toi, sur tes process et tes chiffres, avant de construire quoi que ce soit : c'est l'objet du [cadrage](#).



12. Questions fréquentes



Combien de temps faut-il pour faire passer une tâche dans l'arbre ?

Quelques minutes, à condition de connaître le temps réellement passé. La plupart des tâches sortent à la première ou à la deuxième question. Ce qui prend du temps n'est pas l'arbre, c'est la mesure des durées en amont, à faire sur deux semaines pour éviter de décider sur une impression.

Que faire si je ne sais pas combien de temps prend la tâche ?

Ne réponds pas à la question 4 au jugé. Note les durées pendant deux semaines, sur un carnet ou dans un tableur, en distinguant chaque passage. La plupart des gens découvrent à ce moment que la tâche qui les agace le plus n'est pas celle qui leur coûte le plus cher.

L'IA change-t-elle les réponses de cet arbre ?

Elle déplace une seule question, la troisième. Un modèle de langage traite des cas où les règles ne sont pas entièrement écrites, ce qu'une automatisation classique ne sait pas faire. Elle ne change rien aux quatre autres : une tâche inutile, non écrite, peu coûteuse ou mal alimentée reste un mauvais candidat.

Une tâche qui sort de l'arbre est-elle définitivement écartée ?

Non. Une sortie est un ordre de priorité, pas un refus. Une tâche qui sort à la question 2 devient candidate dès que son mode opératoire



est écrit. Une tâche qui sort à la question 4 le devient si le volume

B Blyx Refais la descente une fois par an, ou après tout changement d'organisation.



Comment savoir si une tâche demande vraiment du jugement ?

Fais faire la même tâche à deux personnes compétentes sur le même dossier et compare les résultats. Si l'écart porte sur la forme, la tâche est automatisable. S'il porte sur la décision elle-même, sur un prix ou sur une relation client, le jugement est bien la valeur ajoutée, et il reste humain.

Qui doit répondre à ces questions dans l'entreprise ?

Celui qui fait la tâche, pas celui qui la commande. Le dirigeant connaît l'intention, l'opérateur connaît les exceptions, et ce sont les exceptions qui décident du coût réel d'un projet. Une descente d'arbre faite depuis un bureau, sans les intéressés, produit des réponses fausses aux questions 2 et 5.

Cet arbre s'applique-t-il aux tâches déjà faites par un logiciel métier ?

Oui, et il donne souvent une sortie inattendue. Beaucoup de tâches manuelles existent pour compenser un logiciel mal paramétré ou mal utilisé. Avant de développer une automatisation par dessus, vérifie que la fonction n'existe pas déjà dans l'outil que tu paies chaque mois.

Faut-il refaire l'arbre après une automatisation livrée ?



Oui, six mois après la mise en production. Les quatre questions du test d'acceptation se reposent à l'identique : qui contrôle, comment une erreur se voit, qui répare, qui possède les accès. C'est le moment où l'on découvre qu'une automatisation tourne encore mais que plus personne ne regarde ce qu'elle produit.

Blyx — cartographie des process & automatisation · blyx-agency.com

