

Si les logiciels de gestion d'entrepôt (WMS) ont atteint une certaine maturité fonctionnelle, ils ont encore de gros progrès à faire sur le plan de l'ergonomie, avec de sérieux gains de productivité à la clef. Récit truculent d'un consultant terrain aguerri sur les mille et une tracasseries subies par les opérateurs en entrepôt faute de dialogue avec les développeurs...

Les WMS vus du terrain



Gilles Fleury
Colise
gfleury@colise.com

66

On ne pouvait pas en dire autant il y a 20 ans, mais aujourd'hui, en 2007, beaucoup des responsables d'entrepôts de taille moyenne¹, ont fait installer ou envisagent de mettre en place un système informatique de gestion – le WMS de nos amis anglo-saxons. Avec le développement de cette demande, le marché offre désormais, un ensemble de solutions dont la majorité a atteint une certaine maturité fonctionnelle avec quelques disparités liées aux cibles visées : telle solution – conçue à l'origine pour les grands entrepôts en palettes entières – offre des modules évolués pour la gestion des mouvements caristes, telle autre – visant le marché de la préparation au détail – est plus riche en modes de préparation et propose un module de précolisage plus précis. Mais, à ces disparités près, l'ensemble de la chaîne d'information, analysée par les concepteurs de haut niveau en collaboration avec les directions logistiques des clients, est traitée complètement. Tout le monde est content...

... enfin, presque tout le monde. Car si l'on veut regarder de plus près par un suivi, vécu sur le terrain, parmi les véritables utilisateurs que sont les opérateurs dans l'entrepôt, il apparaît que leur confort, et par là même leur

productivité, seraient largement améliorés par une foule de mesures, de perfectionnements, d'aménagements. Mais ces aspects, considérés comme des détails relevant de l'implémentation, sont très rarement abordés par les analystes de haut niveau et presque jamais requis par les décideurs des maîtres d'ouvrage.

Il y a pourtant d'importants gains de productivité à réaliser autour de ces aspects ergonomiques, mais les mesures correspondantes échappent à l'analyse globale, il faudrait qu'elles soient négociées entre les 2 maillons extrêmes de la chaîne : le développeur d'un côté, et l'opérateur de l'autre. Malheureusement, ces deux là se rencontrent bien peu, se comprennent encore moins et il y a souvent bien loin de la coupe imaginée par les concepteurs des logiciels aux lèvres de leurs utilisateurs.

Une mauvaise remontée des problèmes

Le premier problème est que beaucoup de ces pertes de productivité, liées à des lenteurs dans l'utilisation ou à la répétition d'erreurs nécessitant des opérations de correction, ne sont même pas connues des éditeurs car elles sont ignorées de leurs interlocuteurs clients pour la bonne raison qu'elles ne sont pas exprimées par les utilisateurs réels, c'est-à-dire les exé-

¹ Disons, à partir de 10 personnes dans l'entrepôt



**“Faire des économies
sur nos coûts de transport,
aujourd’hui c’est majeur !
Non ?”**

Caroline V. Directrice logistique

UN PEU D'AIR © PhotoAlto

Réduire significativement les coûts de transport, gagner en productivité, améliorer la visibilité des commandes de bout en bout, qui peut aujourd’hui se passer de logiciels de TMS ? DDS Shipper, DDS Freight : nos clients chargeurs et prestataires sont la meilleure preuve de l’efficacité de nos solutions.

www.ddslogistics.com



Leader des progiciels de transport et de commerce international

cutants qui exploitent le système. Il y a à cela plusieurs raisons parmi lesquelles des causes sociales profondes. Même si, aujourd'hui, les relations hiérarchiques semblent s'être décrispées par rapport à la grande époque où le Patron tout-puissant, habitant un château et secondé par Monsieur l'Ingénieur en col empesé régnait sur son prolétariat, il serait vain de croire que la communication entre les employés « de base » et les cadres soit aussi simple et directe que les managers voudraient bien le penser. Le tutoiement n'est pas forcément synonyme de sincérité (pas plus dans un sens que dans l'autre d'ailleurs !) et nombre de « sans grades » considèrent comme leur ultime espace de liberté le fait de garder pour eux leurs propres opinions, laissant aux chefs la charge de se dépatouiller avec leurs histoires de productivité. L'opérateur opère donc – puisqu'il est payé pour ça – mais peut lui chaut l'outil qu'on lui impose ; bon ou mauvais, il s'en accommode sans broncher jusqu'à la pause. Là, clope au bec et sourire en coin, il en oublie jusqu'aux rudiments de son art et à la sortie, il commencera sa deuxième journée, l'esprit tranquille.

D'impressionnants stratagèmes de contournement

Au-delà de cette attitude « anarcho-passiviste », l'absence de remontée des problèmes peut être aussi due à la crainte qu'un employé, en difficulté devant l'utilisation d'un système qui lui a été présenté comme miraculeux, soit taxé d'incompétence par ses supérieurs hiérarchiques. Cette appréhension conduit à élaborer des stratégies de contournement qui peuvent être impressionnantes.

Je me souviens avoir cherché des semaines entières l'origine d'inconsistances dans une base de données de réception : des quantités incohérentes entre les réceptions et les charges. Je me suis aperçu finalement que le réceptionniste, que l'applica-



tion n'autorisait pas à corriger a posteriori les quantités reçues, avait réussi à percer les sécurités pour effectuer directement les ajustements dans la base de données. Ce cas, car c'en était un, exceptionnel ô combien, avait été jusqu'à s'offrir sur ses propres deniers, un manuel SQL qu'il potassait le soir. Le matin, il mettait à profit ses connaissances toutes neuves pour modeler les données selon ses exigences. Que dire sinon qu'il n'avait même pas songé à dénoncer une raideur pourtant inacceptable de l'application, jugeant comme hautement improbable qu'on eût pu le prendre au sérieux et comme carrément miraculeux que le problème eût été résolu par un éditeur pour lui inaccessible.

Petits arrangements entre amis

Sans aller jusqu'à cet extrême remarquable – et même admirable – la pratique du terrain montre que les utilisateurs négligent fréquemment de signaler les dysfonctionnements qu'ils subissent usant alors des strata-

gèmes les plus divers pour masquer ou contourner les conséquences de ces défauts. Combien de fois n'a-t-on vu des tricheries sur les paramétrages d'emplacements pour permettre des rangements injustement refusés par le système, ou des bricolages sur les quantités pour autoriser des prélèvements sur des charges autres que celles choisies par le logiciel. C'est véniel certes, mais, outre la perte de temps que ces petits arrangements avec le système engendrent – je rappelle que nous sommes au chapitre « productivité » – ces habitudes conduisent aussi les utilisateurs à blouser la machine même lorsqu'elle a raison (ce qui arrive malgré tout !). C'est ainsi que j'ai découvert un palettier en surcharge alarmante dans un entrepôt où les utilisateurs avaient pris l'habitude de forcer les adresses de rangement, lassés qu'ils étaient des refus du système. Sauf que cette fois, le logiciel détectait effectivement un excès de poids et refusait les rangements, mais personne ne le croyait plus. A la décharge des caristes, il faut dire – mais nous en parlerons plus tard – que le motif du refus n'était pas explicite.

A ce sujet, il est amusant de constater que, devant le comportement complexe, pour ne pas dire franchement erratique de certains logiciels, les utilisateurs ont tendance à leur prêter vie : une autonomie faisant de la machine un interlocuteur à part entière autorisant ainsi des formes de conversations totalement surréalistes : « *il ne veut pas me rentrer ma palette* » ou « *elle m'a jeté sur cette rafale* », les gars utilisant volontiers les pronoms féminins, et les filles le masculin pour désigner un objet pourtant résolument neutre. J'ignore si l'esprit de contradiction systématique de la machine rappelle à certains l'attitude d'un proche, d'un collègue, d'un conjoint², toujours est-il que ce comportement vis-à-vis de ce qui n'est malgré tout qu'un tas de transistors, s'il conduit à

² Toute ressemblance avec une personne existante ou ayant existé etc. etc.

des situations cocasses – les archives de services après-vente informatique regorgent de perles savoureuses – n'en révèle pas moins l'ignorance de l'utilisateur sur le fait qu'un programme, c'est écrit par des gens, avec des bras, des jambes, une tête éventuellement agrémentée d'une paire d'oreilles, et que, très occasionnellement, on peut même leur parler.

Bel effort ! Mais est-ce pour cela que ces utilisateurs courageux seront entendus et, pourquoi pas, écoutés ? Rien de moins sûr.

Un penchant au repli sur soi des développeurs

C'est que la conception et la réalisation d'un logiciel aussi complexe et volumineux qu'un WMS requièrent une énergie et une concentration réellement importantes. Outre que certaines notions à mettre en œuvre ne sont pas si simples à modéliser qu'il y paraisse, le nombre d'objets à gérer, en interaction les uns avec les autres, complique sérieusement l'exercice. Mais surtout, par rapport aux autres



systèmes d'information de l'entreprise manipulant des notions abstraites, dont bien peu savent affirmer la véracité, le fonctionnement d'un

WMS est sans cesse confronté à la dure réalité du physique qu'il est supposé modéliser : lorsque le système prétend qu'une palette comporte maintenant 8 cartons, n'importe quel primate ayant atteint le niveau CM1 est susceptible de le challenger. Le pauvre développeur est donc sans cesse marqué à la culotte, tiré de ses rêves, encerclé qu'il est par une foule agressive d'objets – bien réels ceux-là – dont l'existence et le décompte interpellent en permanence ses plus gracieuses constructions logicielles. Cette désillusion, s'ajoute à la cruelle expérience que fit notre homme lorsque, au sortir d'une école – temple du rationnel – où l'on lui inculqua la rigueur mathématique de l'algorithmique, il se colleta dans la vraie vie avec des environnements d'exploitation délirants, qui tombent en léthargie profonde pendant de longues minutes dès lors que vous avez le malheur d'insérer un CD-ROM, ou pire, un DVD dans un lecteur, sans parler des black-out sur accès réseau et autres vulgarités technologiques de la même farine. Ce n'est certes pas avec de tels exemples qu'on apprend aux informaticiens à respecter leurs utilisateurs. Ajoutez-y des environnements de développement de plus en plus gros, mouvants – quand ils ne sont pas instables, l'émergence constante de nouveaux sabirs de programmation, d'outils toujours plus performants mais de moins en moins contrôlables et voilà en fin de compte notre développeur apeuré, pondant une application dont la majorité du code exécutable aura été écrit par d'autres et dont il devra assumer les bogues en plus de celles qu'il aura lui-même produites. On comprend bien que la maîtrise de tout ce fatras technologique l'éloigne du fonctionnel et nécessite une importante concentration sur le « comment » avec comme conséquence une forte tendance à oublier le « pourquoi » et surtout le « pour qui ». D'où un penchant au repli sur soi confinant par-

Tout un programme !

Un programmeur trouve une grenouille, il la ramasse, l'examine, la met dans sa poche et continue son chemin. Peu après, la grenouille s'agite, le programmeur la prend dans sa main et là, la grenouille se met à parler et dit – « *Je ne suis pas une grenouille, je suis une princesse. Si tu m'embrasses, je redeviendrai la BELLE princesse que j'étais* ». Le programmeur réfléchit, remet la grenouille dans sa poche et continue son chemin. La grenouille remue de plus belle, il la prend de nouveau dans sa main. La grenouille – « *Tu ne m'as pas bien entendu, je ne suis PAS une grenouille, je suis une princesse, et si tu m'embrasses je redeviendrai la SUPER TOP SEXY princesse que j'étais* » – Mais le programmeur la remet dans sa poche et continue son chemin. La grenouille s'agite comme une folle, il la reprend dans sa main.

La grenouille – « *Je ne suis pas une grenouille, je suis une princesse, si tu m'embrasses je redeviendrai la SUPER TOP SEXY princesse que j'étais et je te ferai TOUT ce que tu voudras* » – Le programmeur la regarde et répond – « *je suis bien trop occupé pour faire la fête avec une princesse, mais une grenouille qui parle, ça c'est cool* ».



fois à l'autisme, illustré par de célèbres blagues à base de programmeur, de princesse et de grenouille (voir encadré p.68).

Mais laissons pour un temps nos compères dos à dos, l'un en blouse grise maugréant dans son coin, en bisbille avec sa souris, l'autre en chemise hawaïenne penché sur son écran

Dans leur argumentation commerciale, beaucoup d'éditeurs ont tendance, lorsqu'on leur parle « ergonomie », à répondre « écrans graphiques, multi-fenêtrage, souris ». Ce n'est malheureusement pas aussi simple. Il est indiscutable qu'une présentation graphique bien conçue, avec l'usage d'une bonne symbolique

progrès ont été réalisés par les éditeurs qui offrent à l'utilisateur la possibilité – pour les écrans de listes – de choisir lui-même les colonnes qu'il veut afficher et l'ordre dans lequel les lignes sont classées. Mais il reste encore des systèmes dont les écrans sont figés et pas mal d'améliorations potentielles, par exemple dans la défi-



à la recherche d'un nid de bogues, tout en taquinant sa grenouille, et penchons-nous sur quelques aspects concrets de notre problème.

D'indispensables mesures d'hygiène fonctionnelle

Dans ce qui va suivre, on ne parlera finalement que de détails, de petites choses, mais – et c'est ce qui fait toute la différence entre les tests effectués en laboratoire et l'utilisation réelle d'un système en entrepôt – c'est la répétition de ces petits gains sur un très grand nombre d'opérations, ou sur de gros volumes de données, qui justifie l'effort mis en œuvre dans ces améliorations. Une seconde gagnée sur une opération de picking, dans un entrepôt traitant plusieurs dizaines de milliers de prélèvements par jour, représente au bout du compte des heures d'économie³, moins de fatigue pour les opérateurs, donc moins d'erreur, etc. D'où l'importance de toutes ces petites mesures d'hygiène fonctionnelle, que n'osent pas réclamer les opérateurs, et auxquelles ne pensent pas les développeurs.

³ De l'ordre d'un poste pour 30 000 lignes

favorise une exploitation intuitive maintenant largement passée dans les mœurs. Mais pour une utilisation professionnelle répétitive et continue, la combinaison fenêtre + souris n'est pas favorable à la productivité. L'opérateur doit quitter le clavier pour attraper sa souris, localiser le pointeur, le placer dans la bonne zone, cliquer, lâcher la souris pour le clavier, etc. Pour des saisies de masse, et on verra qu'il peut y en avoir, ce va-et-vient casse littéralement le rythme. Dans les écrans bien conçus, une utilisation intelligente des touches « tabulation » et « entrée » permet de pallier ce problème.

N'afficher que les données utiles

Coté affichage, l'utilisateur attend qu'on lui présente, à tous les stades du processus qui l'occupe, toutes les informations dont il a besoin et seulement celles-là. Mais de manière générale, les développeurs, pour tenter de couvrir tous les besoins, ont tendance à présenter un maximum de données à l'opérateur, d'où un effet de noyade. Sur ce plan, de très gros

progrès ont été réalisés par les éditeurs qui offrent à l'utilisateur la possibilité – pour les écrans de listes – de choisir lui-même les colonnes qu'il veut afficher et l'ordre dans lequel les lignes sont classées. Mais il reste encore des systèmes dont les écrans sont figés et pas mal d'améliorations potentielles, par exemple dans la définition des conditions de filtrage des données. Coté menus, avec les enrichissements fonctionnels des logiciels, le nombre d'options accessibles a littéralement explosé et il n'est pas rare de compter plus de 150 options accessibles à travers des arborescences parfois touffues, alors qu'un utilisateur moyen n'en n'utilise guère plus de 10. Les éditeurs les plus souples bottent en touche en offrant des possibilités de configuration des menus par utilisateur, mais il est rare que l'intégrateur prenne la peine de s'y atteler.

Il y a aussi des détails dans l'affichage qui peuvent améliorer la vie de l'utilisateur comme la simplification des valeurs affichées. Par exemple, si un code article est défini sur 10 chiffres justifié par des zéros à gauche, quel intérêt a-t-on d'afficher les zéros non significatifs puisque ce sont les chiffres de droite que cherche l'opérateur ? N'est-il pas plus simple de lire et de distinguer les chaînes : 87369 plutôt que 0000087369

On imagine bien, dans le deuxième cas, l'opérateur, crayon en main, pointant sur l'écran les zéros afin de

distinguer les chiffres significatifs des simples excipients. Encore une fois, ce qui peut apparaître ici comme un détail, n'en n'est plus un dès lors que des dizaines de milliers d'opérations de ce type sont effectuées chaque jour et tout ce qui peut aider l'œil à dégager la partie utile de l'information est source de productivité nette (meilleure productivité brute et moins d'erreurs).

Des terminaux radio bien pensés

Il est un autre domaine où la présentation des informations mériterait d'être travaillée, c'est celui des terminaux radio, d'autant qu'ils servent en général pour des opérations de

les contraintes du terrain, ambiance sonore, luminosité, instabilité de position, manque de place, les aspects ergonomiques sont essentiels. Dans le choix du matériel d'abord : terminaux légers, avec peu de touches de grande dimension, un grand écran lumineux ; et dans le choix des programmes : peu d'informations mais présentées en gros caractères, limitation des saisies clavier au strict nécessaire et seulement pour des valeurs numériques. Et pourtant, combien ai-je vu de préparateurs, trimbarrant des engins dignes de pianos à queue, les yeux plissés dans la pénombre des allées, en train de jongler avec des touches « shift » et « ctrl », de la taille d'une tête

l'identification automatique. Mais il y a des processus pour lesquels les saisies manuelles sont inévitables.

Il y a d'abord tout ce qui concerne le remplissage des masques de filtrage : « chercher les colis contenant le produit 0000087369 préparés ce matin », ou « quels sont les derniers mouvements réalisés sur la palette 0000007369 » ? Là encore, la saisie de la seule partie significative représente un gain de productivité et diminue les erreurs. L'utilisation de jokers comme le « % » ou les « ? » est un palliatif utile mais plus coûteux en temps de recherche.

Lorsque l'on utilise de manière intensive un WMS, on a souvent l'impression irritante de répéter tou-



masse, répétées plusieurs milliers de fois par jour. Pourtant, les données y sont rarement bien présentées : affichées en trop petits caractères, avec trop d'informations inutiles – l'opérateur s'intéresse en général à une seule donnée à la fois. J'ai même vu un système – et non des moindres – où la position de la quantité à prélever variait avec la longueur du libellé article : imaginez donc la fatigue de l'opérateur, perpétuellement à la recherche du nombre qui l'intéresse le plus dans son travail. Les développeurs avancent des arguments technologiques – contrainte de l'émulation de terminal, taille des écrans – mais a-t-on mesuré le coût des pertes de productivité qui en découlent ? Pour ces équipements, utilisés avec

d'épingle, qui pour saisir un numéro de lot, qui pour réactiver une fonction d'inventaire plantée à chaque fluctuation du réseau ?

Faciliter les saisies indispensables

Pour ce qui est des saisies, le propos du WMS est justement de les limiter en vertu du principe selon lequel une saisie manuelle est une source potentielle d'erreur et un gouffre pour la productivité (de l'ordre 2 caractères/seconde avec un taux d'erreur de 0,3 % pour une excellente dactylo, environ 0,5 car/sec et un taux d'erreur de plusieurs % pour un réceptionniste en fin de journée). D'où l'utilisation massive des lectures par codes à barres, RFID et autres merveilles de

jours la même opération : recherche d'un produit par son code, recherche des charges concernées, saisie d'un écart, ou mouvement de charge etc. La mémorisation et l'édition d'un jeu de requêtes seraient, sur ce plan, un outil apprécié de notre homme en blouse grise.

Favoriser les opérations de masse

S'il est une opération injustement considérée par les développeurs comme accessoire car fonctionnellement pauvre, mais qui est essentielle dans l'exécution du WMS, c'est bien le paramétrage : définition des produits, des conditionnements, des emplacements, des tâches, etc. Il s'agit d'injecter dans le système en

exploitation réelle, des milliers et des milliers de données plus ou moins hétérogènes. Cette opération, capitale au démarrage et fréquemment sous-évaluée, doit être renouvelée au rythme des évolutions de catalogue, des changements de saison. La possibilité de paramétrer les éléments (pro-

incidents. De fait, lorsque l'on s'écarte du droit chemin, du domaine nominal prévu par les concepteurs et programmé par les développeurs, il est souvent coûteux de « rattraper » la situation. Et, pour l'homme à la blouse grise, ce ne sont pas les occasions de se faire blackbouler par la

physiques (encombrement et poids de la palette, dimensions de l'emplacement et charge admissible, occupation) et logiques (familles de stockage, classe de rotation etc.) et il est normal qu'il déclare forfait si aucun emplacement disponible ne satisfait les critères de stockage liés à la



duits, conditionnements, emplacements) par groupes, par familles, par calculs, ou par tout autre moyen permettant de massifier les saisies est un élément important de productivité pour des activités qui font finalement partie intégrante de l'exploitation. A défaut, on utilise souvent des outils externes – tableur ou autre – et l'on procède par import, perdant ainsi les possibilités de contrôle de cohérence interne qui devraient être fournies par le logiciel d'application.

Dans le même ordre d'idée, il arrive fréquemment que des opérations de masse soient réalisées en exploitation : déplacement d'un ensemble de charges, validation en masse de prélèvements ou d'expéditions. Il est important que ces opérations puissent être lancées dans le système en une seule injection, et pas au goutte à goutte, avec un opérateur vissé de longues heures devant l'écran.

Des incidents qui laissent pantois

Sur le plan fonctionnel encore, il est un domaine qui pose souvent problème aux opérateurs : la gestion des

machine, qui manquent : impossibilité d'intégrer les données issues du niveau 3, refus de réceptionner un produit mal paramétré, d'attribuer une adresse de stockage, de prélever un article, de valider un colis etc. etc. Il y a (presque) toujours une bonne raison informatique à chacun de ces rejets, mais encore faut-il que l'utilisateur soit informé correctement de l'incident. Le cas des imports de commandes issues de l'ERP et rejetées par le WMS est un exemple typique d'incident souterrain, non signalé, pas toujours documenté. Le responsable de préparation doit aller de lui-même fouiller dans divers écrans d'événements pour détecter les incidents, les identifier, effectuer les paramétrages manquants et réintégrer – quand toutefois c'est possible – les commandes dans le système. Sur certains sites, il s'est avéré, au terme d'une analyse réaliste, que cette tâche représentait un poste à mi-temps !

Autre grand classique de l'incident : le refus d'attribuer un emplacement à une charge : palette ou colis. Le WMS calcule l'adresse de stockage de la charge en fonction de critères

charge. L'homme à la grenouille nous explique que pour une sélection multi-critères, il est difficile de savoir quel est celui qui n'est pas satisfait, mais il laisse ce faisant le taiseux à la blouse grise face à ses interrogations, désarmé devant les piles de palettes qui s'amoncellent sur le quai. Et le réceptionniste de développer – en fin limier – des heuristiques plus ou moins élaborées pour dénicher le paramètre qui coince et libérer enfin la palette en attente. Mais que de temps perdu et d'énergie gâchée !

On pourrait continuer ainsi à l'envie tant les exemples sont nombreux. Mais les problèmes tournent toujours autour d'une constatation : une meilleure communication avec les utilisateurs finaux, permettrait aux développeurs (je dis bien développeurs et pas analystes), en contact avec le terrain, de mettre au point une foule d'améliorations – petites ou plus importantes – dont le premier impact serait un confort accru des utilisateurs avec pour corollaire une amélioration sensible de la productivité.

Alors, à quand le mariage de la grenouille et de la souris ? ♦