

L'email dans la gestion d'informations personnelles

Julien Thomet
julien.thomet@unifr.ch

Résumé

L'*electronic mail* est devenu le moyen de communication privilégié en entreprise. Son utilisation ne se limite plus à un simple échange asynchrone de messages textuels mais concerne toutes les activités de *personal information management* (*task management*, gestion de documents, recherche, etc.). Nous exposons dans cet article les problèmes engendrés par cette évolution et des pistes pour la conception d'applications de messagerie électronique basées sur les recherches actuelles.

1. INTRODUCTION

Aujourd'hui, l'*email* est devenu le principal moyen de communication dans les entreprises du monde entier, que ce soit au niveau interne ou pour l'échange d'informations avec les clients, les partenaires et les fournisseurs. Cela signifie que les informations critiques concernant la marche des affaires sont éparpillées à travers une multitude d'*emails*. Le nombre de messages traités (envoi et réception) par un utilisateur (en entreprise) par jour a été évalué à 156¹. Ces utilisateurs disent passer 19% de leur temps de travail quotidien au traitement de leurs *emails*, sans garantir que les messages importants ont été lus!

Mais comment est exploité l'*email* actuellement? Quels sont les principaux *challenges* de la conception d'applications de messagerie électronique? Cet article présente tout d'abord l'utilisation de l'*email* dans le contexte de la gestion d'information personnelle ainsi que les problèmes qui en découlent. Sont ensuite étudiées diverses pratiques et techniques d'aide à la gestion des *emails* (classement, visualisation, recherche), avant de conclure avec deux visions possible de l'avenir de la messagerie électronique.

1. The Radicati Group, "Addressing Email Chaos : The Email-ManagerTM Solution", <http://www.radicati.com/?p=49>.

2. L'EMAIL AUJOURD'HUI

La messagerie électronique a été développée à l'origine pour permettre une communication asynchrone, mais il ne fait aucun doute que son utilisation actuelle dépasse les limites de *READMAIL* et *SNDMSG*². En effet, l'*email* d'aujourd'hui est bien plus qu'un simple moyen de communication asynchrone et est utilisé pour diverses tâches :

- gestion de tâches : le premier courriel d'une discussion (*thread*) définit les tâches d'un projet tandis que les suivants rendent compte de leur avancement.
- gestion de documents : dans le cadre d'un travail collaboratif, la messagerie électronique est un support simple pour la gestion des versions de documents.
- gestion de contacts : certains courriels contiennent des informations pertinentes concernant l'expéditeur ou les destinataires (numéros de téléphone, poste au sein d'une société, domicile,...).

On peut également avoir envie d'envoyer un *email* à soi-même pour se rappeler ultérieurement une tâche ou prendre note de quelque chose d'important. La messagerie électronique peut également être utilisée comme système de sauvegarde de documents ou simplement pour le transfert de documents d'un endroit à un autre. L'*email* n'a pas été développée pour supporter ces différentes activités de gestion d'information personnelle : on parle de *email overload*[3]³.

Mais l'*email* en plus d'être critique pour la gestion de l'information personnelle présente d'autres problèmes qui ne concernent pas les autres applications *PIM*. L'un de ces problèmes est la gestion de l'information interpersonnelle [4]. Par exemple, l'*email* permet d'impliquer plus d'une personne dans la réalisation d'une tâche. Alors que les autres activités de *PIM* ne gère que l'information que l'on a créée ou que l'on a découverte soi-même, l'*email* est plus complexe.

Premièrement, la gestion des *emails* a une incidence directe sur le travail d'autres personnes. Ainsi, une réponse arrivée en retard peut menacer le travail de quelqu'un d'autre quelque part. Le problème est accentué sachant qu'un message important pour une personne ne le sera pas forcément pour toutes les personnes concernées par le même courriel ; le contexte d'un *email* dépend de l'utilisateur, de ses buts et

2. Les applications *READMAIL* et *SNDMSG* formèrent le premier système de messagerie électronique développé par Ray Tomlinson dans les années soixantes.

3. Le concept d'*email overload* n'est pas à confondre avec le problème de surcharge d'information ou de *inbox clutter*.

de ses intérêts, ce qui complexifie l'organisation de sa messagerie et la gestion de ses tâches.

Toutes ces difficultés se regroupent en deux problèmes principaux : la fragmentation de l'information et la surcharge d'information.

2.1 La fragmentation de l'information

La fragmentation de l'information dans la messagerie électronique est inhérente à l'*email*, à savoir une forme de communication sous forme d'échange de messages techniquement indépendants les uns des autres. Par exemple, quand un projet est initialisé et que les participants communiquent par courriel, un certain nombre de messages sont échangés durant la réalisation. Chaque message peut contenir des informations sur l'avancement du projet, des prises de décisions, des planifications de réunion, des compte-rendu de *meetings*, des questions, etc. Toutes ces informations, si elles ne sont pas regroupées, restent généralement éparpillées dans plusieurs *emails* et deviennent alors difficiles à retrouver à l'aide d'un tri et d'un outil de recherche textuelle.

2.2 La surcharge d'information

La popularité et la simplicité d'utilisation de l'*email* en ont fait un outil très utilisé. Actuellement, les utilisateurs travaillant en entreprise reçoivent et envoient en moyenne 156 messages par jour (estimés à 233 en 2012). Parmi eux, un grand pourcentage est indésirable : une étude estime que 68.8% du courrier entrant est du *spam*. Le courrier électronique compte également énormément de *semi-spam*[3].

3. LE CLASSEMENT

Diverses solutions sont envisageables pour regrouper l'information et résoudre le problème de la fragmentation. L'idée la plus simple et la plus répandue est le classement manuel (ou *filng*). Whittaker et Sidner [5] ont fait émerger trois catégories d'utilisateurs concernant le classement du courrier électronique : les *frequent filers*, les *no filers* et les *spring cleaners*. Ces stratégies sont basées sur deux critères : l'utilisation de dossiers et le "nettoyage" journalier de la boîte de réception (*inbox*).

- *frequent filers* : leur but est de réduire au maximum le nombre de messages dans leur boîte de réception. Ils opèrent un balayage journalier de l'*inbox* afin de supprimer ou classer les courriels.
- *spring cleaner* : ils classent le contenu de la boîte de réception tous les 1 à 3 mois, mais possèdent toutefois une boîte de réception surchargée.
- *no filers* : comme leur nom l'indique, ces utilisateurs n'utilisent aucun dossier et laisse tout leur courrier électronique dans l'*inbox*. Leur stratégie pour en réduire le nombre d'*emails* consiste en l'archivage et la suppression périodique de grande quantité de messages.

En termes de gestion des tâches, la stratégie du *frequent filer* permet de considérer la boîte de réception comme une "to do list" contenant uniquement les messages concernant les tâches actuelles. Côté défragmentation de l'information, le classement régulier peut permettre de regrouper les messages par *thread* et/ou par projet, facilitant ainsi la recherche directe d'informations. Mais le classement est un processus

cognitif difficile [5] car pour qu'il soit efficace, l'utilisateur doit entre autre être capable d'anticiper ses besoins futurs. L'une des limitations du *foldering* dont il est question, est la *mapping many to one* qui ne permet pas de classer un message dans plusieurs dossiers à la fois contrairement à une classification par activités ou l'utilisation de *tags*.

En ce qui concerne les *no filers*, la gestion des tâches devient un vrai problème lorsque l'utilisateur ne consulte pas souvent sa messagerie et qu'il en reçoit une grande quantité chaque jour. Il y a alors le risque de passer à côté d'informations critiques.

La stratégie du *spring cleaner* est sans doute la plus problématique. Il a été observé que ces utilisateurs ont une certaine difficulté à classer leur courrier après une certaine période. Si les *frequent filer* connaissent bien leur hiérarchie, les *spring cleaners* risquent, à cause d'une sous-utilisation de leurs dossiers, de créer des doublons et ainsi de refragmenter l'information.

Le coût de ces stratégies est évidemment différent, mais bien que la stratégie du *frequent filer* paraissent moins coûteuse en temps de recherche et en localisation d'information, les *no filers* peuvent utiliser des outils avancés pour la recherche ou la visualisation d'information.

3.1 Le classement assisté

L'un des problèmes du classement manuel est le coût en temps que cela peut engendrer. Des techniques ont été développées pour aider l'utilisateur à classer son courrier suivant divers critères.

3.1.1 Le classement par activités

L'une des solutions étudiées entre autre par Dredze *et al*[1] concerne le classement automatique du courrier électronique par activités. Ces dernières ont plusieurs caractéristiques exploitables : elles possèdent un état ("terminée" ou "en cours"), un but et contiennent généralement des processus (formels ou informels) pour atteindre le but. De plus des activités peuvent être liées entre elles.

Le but est de fournir aux utilisateurs un outil efficace de gestion des activités basé sur l'*email* comme moyen de communication collaboratif. Les algorithmes développés par Dredze *et al*[1]⁴ permettent de classer dynamiquement⁵ les messages en étudiant à la fois les personnes impliquées dans une activité en étudiant ses destinataires et les similarités du contenu avec d'autres *emails*. Les résultats obtenus font état d'un meilleur taux de rappel que pour la technique du *threading*, mais avec une moins bonne précision ; en d'autres termes, le *threading* est plus précis pour déterminer à quelle activité appartient un *email* alors que les algorithmes de Dredze *et al*[1] identifient plus justement si un message appartient ou non à une activité.

Cependant, le classement orienté activités reste un problème difficile, notamment lorsque le nombre d'activités croît, que les *emails* contiennent des informations concernant plusieurs

4. *SimSubset*, *SimOverlap* et *SimContent*

5. les activités ne sont pas créées automatiquement : l'utilisateur doit initialiser une nouvelle activité manuellement.

activités ou qu'une petite partie du contenu seulement concerne une activité. Outre la classification, l'identification de nouvelles activités est un problème encore plus délicat.

3.1.2 Le classement par threads

La technique précitée du *threading*, et utilisée comme moyen de comparaison pour les algorithmes *SimSubset*, *SimOverlap* et *SimContent* [1], permet de regrouper des messages automatiquement par conversations et, couplé à une visualisation adéquate, de retrouver l'information plus aisément. Le classement se base principalement sur le titre (ou sujet) des *emails* et sur l'ensemble des personnes impliquées dans la discussion. Cependant, des sujets similaires ou au contraire complètement différents (bien qu'identifiants des courriels d'une même discussion) complexifient la classification.

Dans une vision de *task management*, le *threading*, contrairement à la classification précédente, ne permet pas d'identifier précisément des tâches, car un thread ou un *email* correspond rarement à une tâche particulière. Pour cette raison l'idée de "thrask" [4] (mot-valise formé de *thread* et *task*) ou collection personnalisable basée sur les *threads* a été développée; l'utilisateur peut y ajouter des messages non liés à une conversation ou les en retirer, ainsi un *thrask* représente une collection de tâches à la place d'une simple série de messages.

Comme nous l'avons vu, le classement (automatique ou manuel) permet de regrouper l'information fragmentée, mais également de solutionner d'une certaine manière la surcharge d'information au niveau de la boîte de réception (*inbox clutter*). Associé à des systèmes de visualisation idoines, le classement s'avère être encore plus efficace, spécialement pour la localisation d'information.

4. LA VISUALISATION D'INFORMATION

Gruen *et al*[2] ont travaillé sur les différents prototypes de *ReMail*⁶, un système de messagerie électronique "réinventant l'*email*"⁷. Dès le second prototype, *ReMail* présentait une technique de visualisation des discussions appelée *thread map*. L'idée était de représenter de façon intuitive une séquence de courriels formant une discussion.

La figure 1 montre un exemple de *thread map* dans le dernier prototype de *ReMail*. Concrètement, chaque noeuds représente un courriel, le plus grand correspondant au *mail* courant. Chaque noeud est cliquable et renvoie au message correspondant. Les noeuds sont organisés chronologiquement de gauche à droite. Les différents types de noeuds donnent une indication sur l'importance de l'expéditeur (un cercle vide peut représenter un contact important), sur le statut du message (lu ou non-lu) ou encore sur l'intervalle de temps dans lequel les différents messages ont été envoyés; le dernier prototype offre à l'utilisateur la possibilité de choisir le mode de visualisation. Finalement, ce diagramme comprend des *thread arcs* de type "réponds à" connectant les noeuds entre eux.

Ce type de visualisation permet de remettre un message dans un contexte de conversation s'échelonnant dans le temps et

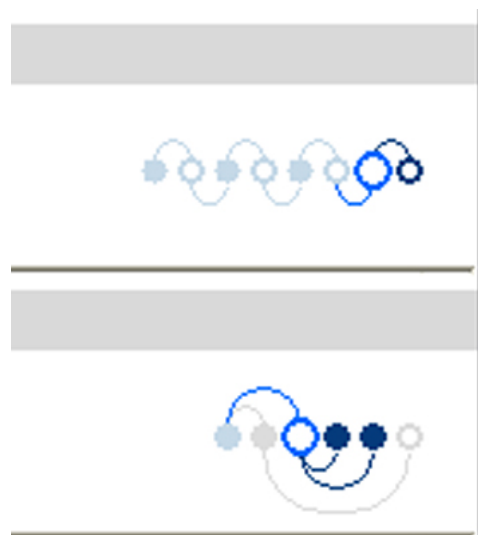


Figure 1: *Thread map* : visualisation des *threads* dans *ReMail*

de faciliter la navigation à l'intérieur d'une conversation. En réponse au *inbox clutter*, *ReMail* offre également la possibilité de visualiser les *threads* sous forme de liste en affichant uniquement le dernier message comme représentant de la discussion et en précisant dans une colonne voisine le nombre de message échangés. Cette technique de suppression permet de réduire considérablement le nombre d'*items*.

ReMail offre bien d'autres fonctionnalités (extraction de dates, *message map*, etc.), mais c'est particulièrement les deux visualisations présentées qui ont été appréciées par les utilisateurs lors des tests en témoignant également des implémentations de *thread map* dans *GNUMail*⁸ et *Thunderbird*^{9 10}

5. LA RECHERCHE D'INFORMATION

Jusqu'ici nous avons considéré l'organisation de l'information ainsi que des techniques de visualisation permettant de mieux se représenter un message dans son contexte. Mais l'une des tâches récurrentes dans la gestion des *emails* est la recherche efficace d'information. L'organisation de sa messagerie offre déjà une méthode qui permet de restreindre rapidement les domaines de recherche pour une information précise. Comme pour l'organisation, la recherche d'information se fait en plusieurs étapes mais pas toujours dans les meilleures conditions. William Jones [3] évoque quatre problèmes globaux concernant la recherche d'information.

Le premier problème est que l'utilisateur oublie; même en utilisant toutes les outils d'un système de messagerie électronique moderne (*tags*, alertes, dossiers, etc.), un message peut rester sans réponse jusqu'à ce qu'il soit trop tard. Les alertes deviennent tellement nombreuses que l'on n'y fait plus attention ou que l'on ne les utilise pas comme outil de

8. <http://collaboration-world.com/gnumail>

9. <http://www.mozilla-europe.org/fr/products/thunderbird/>

10. <https://addons.mozilla.org/fr/thunderbird/addon/6533>

6. <http://www.research.ibm.com/remail/>

7. *ReMail = Reinventing Email*

rappel. Whittaker *et al*[5] évoquent également le comportement d'utilisateurs ayant créé un dossier "à faire", mais qu'ils oublient de consulter régulièrement.

Pour rechercher une information précise, il faut soit savoir exactement où elle se trouve ou connaître une partie de l'information souhaitée (date, sujet, expéditeur) et utiliser des outils adéquats pour y accéder (recherche textuelle, tri, etc.). Mais avec une surcharge constante de messages, il est parfois difficile de se remémorer une information partiellement. Il peut également arriver que le courriel désiré soit devant nos yeux, mais que nous ne le remarquons pas parce qu'on ne reconnaît pas le message qui comporte peut-être un sujet non significatif.

Lors des tests utilisateurs de *ReMail* [2], il a été observé, sans surprise, que l'une des fonctionnalités les plus utilisées dans le dernier prototype était la recherche textuelle (*full-text searching*). Les utilisateurs appréciaient également la recherche instantanée (*instant search*) ainsi que la recherche de dates (basée sur l'extraction de dates). Whittaker *et al*[4] évoquent également les trois stratégies les plus utilisées pour l'accès à l'information dans un système de messagerie électronique : "*folders [...], search and sort*".

Comme mentionné précédemment, la classement manuel est un processus cognitif difficile et c'est pourquoi certains utilisateurs abandonnent le classement au profit d'outils de recherche élaborés (par exemple *Google Desktop*¹¹) qui facilitent considérablement la recherche d'informations. Ces outils résolvent partiellement le problème de la fragmentation pas uniquement au niveau des *emails* mais plus globalement pour toute l'information à laquelle à accès un utilisateur. Néanmoins, la formulation de requêtes ciblées peut être aussi difficile que la classification par dossiers. Quant aux fonctionnalités de tri, implémentées dans tous les logiciels de messagerie, elles permettent d'utiliser la capacité d'un utilisateur à se rappeler partiellement le contenu d'un message, mais reste un moyen indirect de rechercher l'information.

Finalement, avec une boîte de réception surchargée, lorsque l'on quitte notre messagerie sommes-nous sûr que toutes les tâches importantes ont été effectuées? Sachant que les tâches sont éparpillées dans différents messages, peut-être dans différents dossiers, s'assurer que tout est réglé n'est pas si anodin.

6. ET APRÈS ?

Nous avons présenté les principaux problèmes liés à la gestion des *emails*, certains comportements des utilisateurs et des solutions étudiées pour résoudre les deux problèmes principaux de la gestion d'information que sont la fragmentation et la surcharge d'information.

Mais alors, quels sont les scénarios possibles concernant la gestion du courrier électronique? William Jones dans un chapitre intitulé *Email disappears?*[3] entrevoit deux visions possibles : l'*email* se relève ou meurt. Il est important de préciser que l'on considère ici l'*email* comme une interaction et non comme une technologie.

6.1 Le règne de l'email

Dans cette première vision du futur, l'*email* est utilisé pour toutes les tâches citées dans les premières sections; on utilise l'*email* pour gérer ses tâches, ses documents et ses contacts. On utilise l'*email* pour tout puisque c'est dans sa messagerie que l'on passe le plus de temps. Pourquoi rechercher l'adresse d'un contact dans une base de données alors qu'il suffit de retrouver un ancien mail de cette personne? Pourquoi créer une nouvelle tâche dans un module dédié alors qu'un courriel peut servir d'enregistrement et le statut du message fournir des indications sur l'avancement de la tâche?

Deux inconvénients dès lors apparaissent : les applications actuelles sont déjà surchargées de modules différents et en ajouter ne ferait qu'empirer les choses. Deuxièmement, les utilisateurs doivent ressentir le besoin de fermer leur boîte de réception pour réellement avancer dans leur travail.

6.2 La mort de l'email

La seconde vision de William Jones s'articule autour de quatre points principaux :

- une représentation externe distribuée,
- *Don't email; communicate*,
- placer la conversation dans son contexte,
- un *Attention Central* permettant de voir d'un coup d'oeil ce qui requiert notre attention.

6.2.1 La représentation externe distribuée

Une représentation externe distribuée, comme son nom l'indique, permet de rassembler de l'information dans une application externe, accessible et enrichissable par tous ou une partie des personnes impliquées dans une tâche. L'exemple pris par William Jones est celui du *wiki*. À la place de défragmenter l'information contenue dans une série d'emails pour avoir un aperçu des décisions prises, un *wiki* fournit un résumé des dernières discussions, de ce qui a été fait et de ce qui reste à faire. Les personnes autorisées peuvent à tout moment enrichir le contenu du *wiki*.

Mais des questions se posent encore. Certes ce genre d'outil est utile pour la gestion d'un projet, l'organisation d'un voyage, mais qu'en est-il pour des la coordination et la négociation d'occasions plus simple pour laquelle une vingtaine d'emails suffiraient? Et même lorsqu'un *wiki* serait envisageable, il est si simple d'envoyer quelques mails... Cependant, les systèmes de *wiki* sont encore à l'état primitif et les avantages d'une représentation distribués sont plus que suffisants pour continuer à explorer l'exploitation de telles plateformes.

6.2.2 Don't email; communicate

Ce point s'articule autour d'une fonctionnalité *Ask/Tell*[3] permettant de choisir parmi divers moyens de communication (*instant messaging, SMS, email, téléphone, etc.*) celui qui paraît le plus adéquat pour communiquer l'information courante.

6.2.3 La conversation dans son contexte

Le problème soulevé est le changement de contexte nécessaire dans la communication par courrier électronique. En

11. <http://desktop.google.com/fr/>

effet, avant d'envoyer un message, l'utilisateur se trouve généralement dans un autre contexte que son application de messagerie. Pour rédiger un *email* concernant les informations courantes, il doit habituellement quitter le contexte courant et démarrer son applicaton de messagerie électronique.

L'idée est de pouvoir utiliser la commande *Ask/Tell* n'importe où, n'importe quand et avec n'importe quelle type d'information. Mais plus encore, Wiliam Jones préconise que l'information qui a déclenché l'utilisation de la commande *Ask/Tell* dans l'esprit de l'utilisateur, garde une trace de la conversation initialisée. Il serait ainsi possible de retrouver l'information déclencheuse et d'être informé de l'état de la conversation, sans faire appel à une messagerie dédiée.

6.2.4 Attention Central

Finalement, un système d'alerte serait utilisé pour avertir l'utilisateur qu'il y a des choses qui requiert son attention, par exemple une tâche à accomplir, un rendez-vous, une information importante à communiquer, etc.

7. CONCLUSION

L'*email* est actuellement l'un des moyen de communication les plus utilisés de part sa facilité d'utilisation et son caractère asynchrone. Mais sa popularité est aussi l'une de ses menaces ; le nombre de courriels échangés quotidiennement ne cesse d'augmenter, dont une bonne partie n'est d'aucune utilité. Cette surcharge d'information s'accompagne d'une fragmentation de l'information, inhérente à l'*email*.

À travers cet article, nous avons identifié les problèmes sous-jacents et présenté les solutions de divers auteurs, du classement automatique du courrier électronique à la visualisation contextuelle des *emails*, pour terminer avec des solutions plus globales remettant en question la place de la messagerie électronique dans la gestion de l'information personnelle

et interpersonnelle.

Il ne fait aucun doute que l'*email* a encore de beaux jours devant lui, du moins en tant que technologie de communication asynchrone. Wiliam Jones [3] affirme qu'en quarante ans, les logiciels de messagerie électronique ont très peu évolué. Il est à espérer que le même constat ne sera pas le même dans quarante ans, auquel cas, la surcharge d'information aura atteint des sommets.

8. REFERENCES

- [1] Mark Dredze, Tessa Lau, and Nicholas Kushmerick. Automatically classifying emails into activities. In *IUI '06 : Proceedings of the 11th international conference on Intelligent user interfaces*, pages 70–77, New York, NY, USA, 2006. ACM.
- [2] Daniel Gruen, Steven L. Rohall, Suzanne Minassian, Bernard Kerr, Paul Moody, Bob Stachel, Martin Wattenberg, and Eric Wilcox. Lessons from the remail prototypes. In *CSCW '04 : Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 152–161, New York, NY, USA, 2004. ACM.
- [3] William Jones. *Keeping Found Things Found : The Study and Practice of Personal Information Management (Interactive Technologies) (Interactive Technologies)*, pages 272–297. Morgan Kaufmann, November 2007.
- [4] Steve Whittaker, Victoria Bellotti, and Jacek Gwizdka. Email in personal information management. *Commun. ACM*, 49(1) :68–73, 2006.
- [5] Steve Whittaker and Candace Sidner. Email overload : exploring personal information management of email. In *CHI '96 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 276–283, New York, NY, USA, 1996. ACM Press.