

# Visualisation de digests d'emails en entreprise

Romain Vuillemot  
Université de Lyon, CNRS  
INSA-Lyon, LIRIS, UMR5205  
F-69621 Villeurbanne, France  
romain.vuillemot@insa-lyon.fr

Jérôme Mulsant  
Alinto  
Lyon, France  
jmulsant@alinto.net

Gaëlle Recourcé  
Kwaga  
Paris, France  
recource@kwaga.com

## RESUME

Dans cet article nous nous intéressons à la visualisation de *digests* d'emails, à savoir la représentation graphique compacte d'un ensemble d'emails en un seul email [26]. L'objectif de cette technique est de synthétiser un ensemble de données et de tâches contenues dans les emails, afin d'aider l'utilisateur à la prise de décision et à la communication du résultat. Notre contribution consiste en l'extension de cette technique introduite dans [26] en proposant une série détaillée de digests, incluant des visualisations telles qu'un nuage de mots clés ou un graphe. Nous détaillons l'implémentation dans un webmail de test et discutons les premiers problèmes techniques et ergonomiques liés à la mise en production d'une telle technique.

## MOTS CLES

Digest d'email, visualisation, template sémantique.

## ABSTRACT

In this article we focus on emails *digests* visualization, namely the graphical representation of a compact set of emails in a single email [26]. The objective of this technique is to synthesize a set of data contained in emails to assist the user in decision-making and results sharing. Our contribution consists in extending the techniques introduced in [26] by describing digests examples, using visualization techniques such as word clouds or graphs. We detail the implementation as a webmail and we discuss our early technical and ergonomic issues observed while deploying this technique in a production context.

## Categories and Subject Descriptors

H.5.2 [Information Interfaces And Presentation]: User Interfaces, Graphical user interfaces

## General Terms

Design.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IHM'11, October 24-27, 2011, Sophia Antipolis, France  
Copyright © 2011 ACM 978-1-4503-0822-9/11/10 ...\$10.00.

## Keywords

Email digest, visualization, semantic template.

## 1. INTRODUCTION

L'email est un des systèmes de communication électronique les plus anciens (apparu en 1972), et toujours parmi les plus utilisés. De nombreuses études ou cas d'études montrent qu'il est à la fois socialement accepté et considéré comme fiable en entreprise. Ses atouts sont un protocole de communication ouvert, une architecture décentralisée et la possibilité d'implémenter ou d'utiliser tout type de client. Il n'est donc pas nécessaire d'appartenir à une communauté pour créer une adresse email, et les messages échangés ne sont pas sujets à des changements de termes d'utilisation comme peuvent l'être des réseaux sociaux propriétaires et centralisés [28].

Cependant la simplicité du protocole d'email peut provoquer certains désagréments pour les utilisateurs et diminuer leur productivité. Le premier et bien connu est le SPAM, à savoir la réception de messages non sollicités. Ensuite l'usage de l'email n'imposant pas de processus ou de contrainte d'usage [18, 6] un grand nombre d'emails peut être échangé sous forme hétérogène. Par exemple l'organisation d'un rendez-vous, la sauvegarde de fichiers ou le travail collaboratif sont des tâches courantes effectuées par email mais toutes partent d'un champ texte libre, et finissent par un envoi/réception d'emails ce qui entraîne des phénomènes de surcharge d'email dits d'*email overload* [29]. Enfin l'avènement récent de messages classés en Bacn [1], à savoir des messages qui sont intéressants et sollicités (newsletters, mises à jour sociales, alertes systèmes, etc.), mais que l'on ne souhaite pas traiter tout de suite, provoquent un problème de traitement de masse d'emails nouveaux. Ces traitements ne peuvent être résolus par les techniques de classification ou de résumé d'email [3] car les tâches à réaliser sur ces emails de type Bacn sont très variées et évoluent rapidement au fil du temps. Par exemple une newsletter datant de plusieurs semaines peut posséder un contenu plus ou moins pertinent si l'utilisateur réalise une tâche de veille (étude des évolutions à court terme) ou bien d'exploration d'archive (afin d'étudier des tendances sur le long terme).

### 1.1 Traitement répétitif des emails

Ce dernier type d'emails dits "Bacn" doit être traité manuellement et engendre donc un processus long et fortement répétitif pour l'utilisateur, sans demander pour autant de compétence particulière. Face à sa boîte email, l'utilisateur est alors surchargé par des étapes successives de lecture et

d'extraction de données qui peuvent certes être assistées avec des outils externes (Microsoft Excel, bases de données, etc.), mais le manque d'interopérabilité entraîne de nombreux copier/coller et accroît le risque d'erreur. De surcroît, l'utilisateur doit posséder et configurer ces outils qui ne sont pas forcément disponibles chez tous ses collaborateurs, ce qui limite donc la communication ou l'édition collaborative des résultats.

## 1.2 Exemples de traitement d'emails répétitifs

De nombreux exemples de traitement répétitif d'emails existent. Par exemple organisation d'un rendez-vous uniquement via email engendre au minimum autant d'emails que de participants, voir plus. Les appels à soumissions de conférences scientifiques (Call for Papers, CFP) sont aussi nombreux et doivent être suivis par tout scientifique afin de connaître les nouveaux événements (lieu, dates, comité d'organisation, etc.) organisés par une communauté. Les newsletters sont également très répétitives dans leur mise en forme avec des en-têtes et pieds de page souvent identiques car issus d'un même template. Ce type d'email pouvant même être apparenté à des pages web, l'exploitation de la similarité est possible car celle-ci est relativement stable au fil du temps [9].

## 1.3 Motivation pour l'usage de digests d'emails

Un digest d'email est la représentation graphique compacte d'un ensemble d'emails en un seul email [26]. Le design de ces digests doit être pertinent par rapport aux données ou à la tâche à réaliser, afin de permettre à l'utilisateur de prendre une décision et de la communiquer. L'implémentation actuelle de cette technique consiste en une option dans les listes de diffusion afin de permettre de réduire le nombre d'emails reçus par les clients (exemple de digest issu d'une liste de diffusion figure 1, à gauche). Certains clients mails comme Mozilla Thunderbird permettent aussi la compilation de plusieurs mails en un seul, cette fois côté client, mais le rendu est simpliste (figure 1, à droite). Nous avons déjà proposé d'étendre cette technique à tout type d'emails [26], et d'utiliser des techniques de visualisation existantes afin de produire un nouvel email de synthèse. A notre connaissance, il n'existe pas de système exploitant cette technique généralisée de digest d'emails.



Figure 1: Types de digests actuels. A gauche à partir d'une liste de diffusion. A droite à partir d'un client mail classique (Mozilla Thunderbird).

## 2. INTERFACES ACTUELLES DE VISUALISATION D'EMAIL

### 2.1 Interfaces classiques de gestion d'emails

Les interfaces actuelles de gestion d'emails peuvent être considérées comme des structures visuelles *simples* au regard de [13]. En effet, les emails sont principalement organisés sous forme de liste, ce qui permet de les classer selon leur ordre chronologique d'arrivée ou l'ordre alphabétique des auteurs. Deux autres panneaux complètent généralement cette liste : une vue arborescente de dossiers ou labels, et une vue de contenu d'emails. Ces panneaux forment une vue multiple coordonnée où le choix d'un élément dans un panneau permet d'obtenir plus de détail dans les autres panneaux, suivant un ordre hiérarchique. A noter que Gmail [10] innove en mélangeant la vue de contenu et de liste d'emails par volonté de proposer une interface orientée conversation.

### 2.2 Interfaces externes de visualisation

Selon les multiples facettes que comporte l'email (document textuel, graphe de communication, conversation) autant de techniques de visualisation ont été développées spécifiquement ou adaptées. Les emails peuvent être explorés sous forme de distribution thématique [24], reposant sur des techniques d'extraction et de classification de texte. L'analyse d'archive d'emails [15] permet la compréhension du graphe de communication, à savoir la structure indirectement générée par les échanges d'emails. Ces interfaces ont la particularité d'être *externes* aux emails dans le sens où elles ne permettent pas l'édition et la composition d'un email. Elles provoquent ainsi une discontinuité au niveau du focus de l'utilisateur qui doit jongler entre la vue de travail centrée email et les vues multiples d'analyse et de réflexivité sur son activité.

### 2.3 Interfaces visuelles centrées emails

Les interfaces visuelles centrées email sont souvent des interfaces classiques, dont certaines variables graphiques indiquent des informations sur les propriétés des emails. SPAM Gui affiche le score de la détection du SPAM (caché dans les en-têtes d'emails) sous forme de gradient de couleur [2] afin d'impliquer l'utilisateur dans la boucle de décision d'analyse de SPAM. De même, la profondeur d'une conversation peut être affichée avec un code de couleur [21]. L'ajout de widgets de relecture d'emails [25] permet la relecture d'arrivée d'emails et ainsi revenir à un état passé pour traiter ses emails. Ces approches permettent à l'utilisateur de garder le focus sur ses emails, mais restent relativement limitées.

## 3. EXEMPLE DE GENERATION D'UN DIGEST

Dans cette section nous présentons les cinq étapes permettant la génération d'un digest d'email, telles que nous les envisageons. Cette génération est initiée par l'utilisateur dès qu'il sélectionne deux emails au plus. Elle se termine par l'apparition d'un nouvel email qui contient l'agrégation des emails sélectionnés (figure 2).

- ① Tout d'abord l'utilisateur sélectionne un ensemble de  $N$  messages ( $N \geq 2$ ) au moyen de cases à cocher.
- ② Ensuite un feedback immédiat indique le nombre d'emails sélectionnés.
- ③ L'utilisateur sélectionne le type de digest qu'il souhaite : classique, nuage de mot, carte géographique, etc.

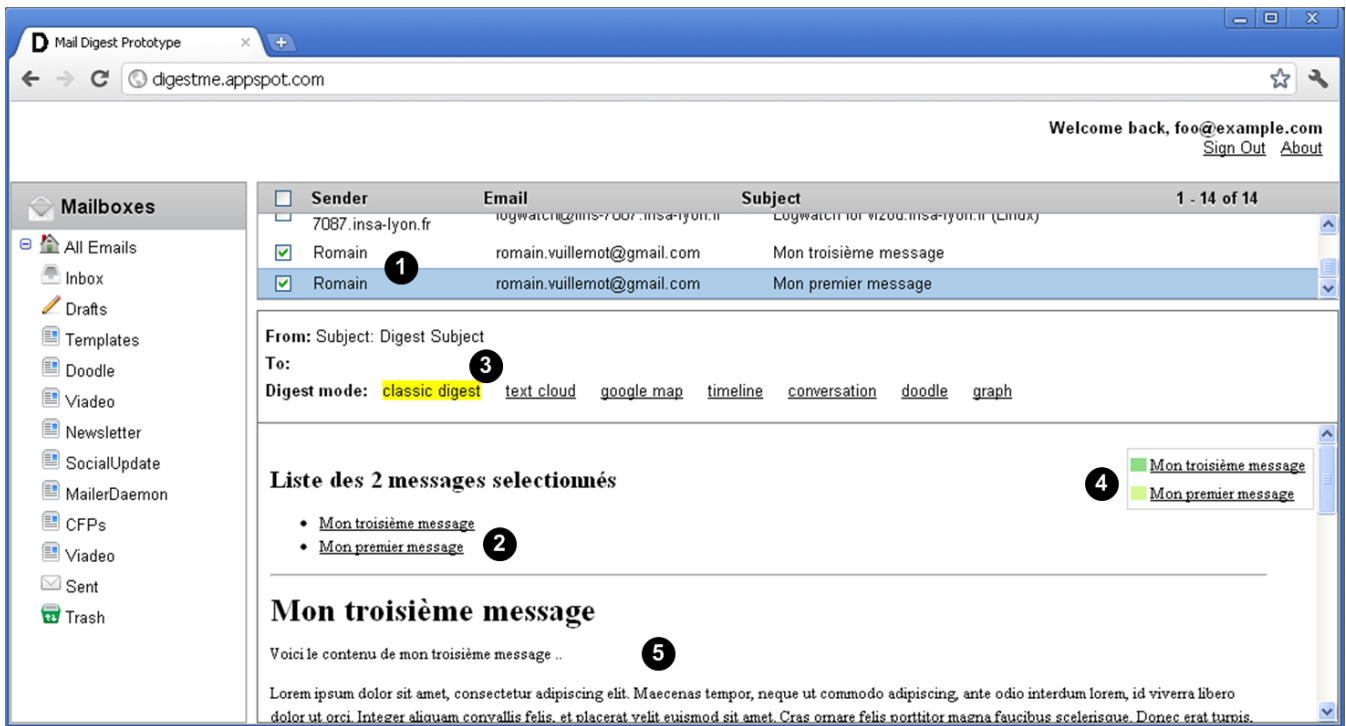


Figure 2: Etapes de génération d'un digest d'email par l'utilisateur.

④ Le digest apparaît accompagné d'une légende interactive permettant de naviguer dans celui-ci et de comprendre l'éventuel code de couleur qu'il contient.

⑤ Le résultat est un email à part entière. Autrement dit une série de caractères au format email encodé en UTF-8, avec en-tête, corps de messages et d'éventuelles pièces jointes.

#### 4. APERÇU DU TRAITEMENT DES DONNÉES

Les étapes précédentes sont déclenchées côté client, mais font appel à des processus de traitement de données côté serveur (figure 3). Avant toute forme de visualisation, les données et tâches doivent être extraites à partir des emails, et ensuite modélisées selon le domaine correspondant (figure 3). Cette phase relève du domaine de l'extraction d'information et est réalisée à partir de règles permettant d'extraire des triplets de données [8], ainsi qu'à partir d'appels au moteur d'analyse sémantique de l'entreprise Kwaga accessible sous forme de service web.

La modélisation et le stockage des données sont ensuite réalisés sous forme de graphe. Cette structure convient bien pour la modélisation de données de type PIM (*Personal Information Management*) [14], ainsi que pour l'agrégation [23] des données. Pour illustrer ces étapes d'extraction et de modélisation, nous prenons le cas simple d'un email d'organisation d'un rendez-vous. Un tel email contient des dates, des personnes et le choix de ces dates par les personnes. Les données seront ainsi par exemple une séquence de dates (figure 4), entre autres. La tâche sera une séquence de choix parmi les dates (figure 5).

Ces données sont stockés au format RDF (*Resource Description Framework*) qui est un standard développé par le

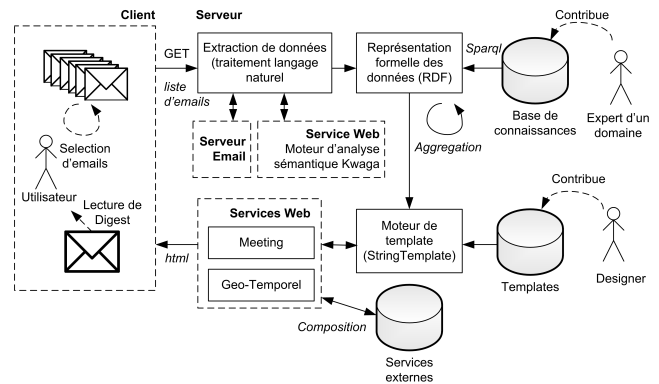


Figure 3: Architecture du traitement des données.

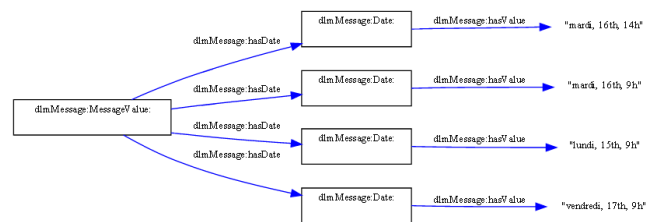


Figure 4: Exemple de graphe de données pour les dates d'un rendez-vous. Le message (MessageValue) est une ressource à laquelle sont associées des dates (Date) et des valeurs de dates.

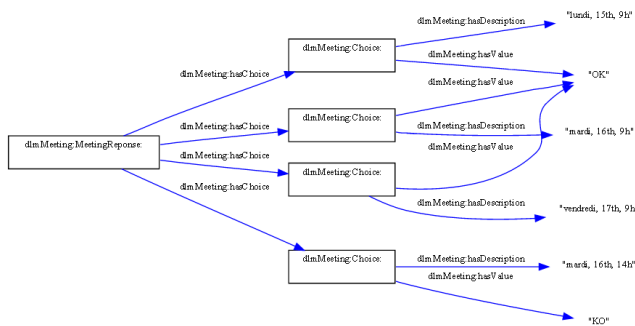


Figure 5: Exemple de graphe de tâche d'un rendez-vous. La tâche (MeetingResponse) est une ressource à laquelle sont associées des choix (Choice) et des valeurs de dates.

## 5. TEMPLATES SEMANTIQUES DE VISUALISATION

Nous introduisons un système de templates permettant le rendu visuel des digests, ainsi que des exemples.

### 5.1 Définition des templates sémantiques

Les templates sont une façon courante de stocker la mise en forme visuelle des données, de manière indépendante des données elles-mêmes [20]. Utiliser cette approche permet d'introduire une certaine flexibilité dans le design des digests qui pourront être ainsi modifiés facilement. Une alternative aurait pu être de définir de manière fixe les digests, en coopérant avec des experts du domaine, afin de proposer un ou plusieurs types de visualisations adaptées à des tâches d'analyse visuelle préalablement identifiées [4]. Mais étant donnée la nature variée du contenu des emails, nous proposons un système général, qui pourra être étendu et même inclure d'autres types de visualisation générées par des services web et embarquées dans les emails. Etendre un template existant consiste à modifier le fichier de définition du template (qui est écrit dans le langage StringTemplate [20] proche du HTML) ce qui demande des compétences techniques assez répandues. La notion de *sémantique* est liée au fait que les données qui sont traitées dans ces templates peuvent être interprétées par le système (comme par exemple les dates, lieu et les personnes) car décrites formellement dans le graphe de données et de tâche.

### 5.2 Exemples de templates de digests

Nous décrivons et motivons différents templates (table 1) qui implémentent des visualisations simples et efficaces, en particulier liés aux exemples précédemment introduits d'organisation de réunion et d'analyse des CFP. Chaque template correspond aux données et tâches extraites dans un domaine précis. La détection automatique du domaine en fonction du type d'email sélectionné par l'utilisateur est une perspective de travail future. Dans le cadre de cet article nous nous intéressons au choix manuel par l'utilisateur, et le système génère le digest associé, qui peut ne pas être celui le plus optimal ou adapté au type de données.

**Digest d'héritage.** Le digest réutilise un design existant et commun aux emails sélectionnés.

Digest type	Description
Héritage	<i>Héritage de design existant</i>
Classique	<i>Séquence classique d'email</i>
Rendez-vous	<i>Compilation de réponses</i>
Nuage de mot	<i>Fréquence des mots</i>
Différence	<i>Différence entre deux textes</i>
Géo-temporel	<i>Lieux et dates</i>
Graphe	<i>Graphe de communication</i>

Table 1: Exemples de templates de digests.

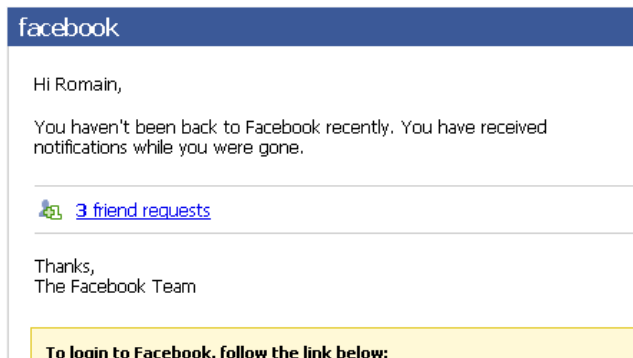


Figure 6: Digest héritant du design d'emails existants.

De nombreux messages contiennent la même mise en forme (newsletters, notifications sociales, etc.), dont le contenu varie peu. Ainsi ce type de digest vise à extraire le contenu variable de ces emails, à l'aggréger, et à l'insérer dans le contenu identique des emails. Dans l'exemple d'une notification sociale (figure 6) il s'agira de mettre à jour le nombre d'amis (dans ce cas 3 amis, qui proviennent de deux emails contenant respectivement 2 et 1 amis). L'opérateur d'agrégation est la somme. Pour cet exemple, il a été nécessaire de recréer manuellement le design original, mais une extraction automatique pourrait être envisagée [19] et l'opérateur d'agrégation déduit au fil de la sélection d'emails. A noter que l'utilisation de design existant peut poser des problèmes légaux [16].

**Digest classique.** Le digest compile les emails de manière séquentielle selon leur ordre de sélection. Une légende latérale permet de naviguer dans cette compilation.

### Liste des 5 messages sélectionnés

- [1st\_CFP] 1st Workshop on Game Accessibility: Xtreme Interaction ■ E-mail 1
- 1st European Workshop on HCI Design and Evaluation (8 April 20) ■ E-mail 2
- [Specif-forum] Fwd: 2nd Conference on Complex Systems Design & ■ E-mail 3
- ■ E-mail 4

Figure 7: Digest classique.

Ce digest (figure 7) reprend le type de digest actuellement implémenté par les listes de diffusion et les clients mail (figure 1) et permet de naviguer de manière continue dans une série d'emails.

**Digest de rendez-vous** Le digest synthétise le résultat de l'organisation d'un rendez-vous.

	march 15th @ 9 a.m	march 16th @ 9 a.m	march 16th @ 2 p.m	march 19th @ 9 a.m
Jean-Marc				
Mohand-Said Hacid	OK		OK	
Romain	OK	OK	OK	
Amin	??	??	??	??

Figure 8: Digest de rendez-vous.

Nous proposons pour cela de réutiliser le même design qu'un Doodle [5] (figure 8). Ce type de digest nécessite une analyse sémantique d'emails (en particulier via l'appel au web service Kwaga) qui permet l'extraction d'entités (personnes et dates), ainsi que les relations entre ces entités (réponse des personnes aux dates). Le résultat du digest peut ensuite être transféré aux personnes concernées par simple transfert d'email. Les données manquantes sont affichées dans le digest sous forme de points d'interrogations. D'autres stratégies de design pourront également communiquer les données incertaines [22].

**Digest de nuage de mots** Le digest génère un nuage de mot dont la taille des mots est leur fréquence dans les emails sélectionnés, et l'ordre leur ordre d'apparition dans la sélection d'emails.



Figure 9: Digest de nuage de mots

Un nuage de mots est un mécanisme simple et efficace permettant d'avoir un aperçu d'une grande quantité de texte. Il offre ainsi une vue globale d'un grand volume d'emails, qui peut être une première étape dans l'analyse visuelle. Son efficacité est particulièrement liée aux mécanismes de nettoyage de texte (mots vides, etc.) mais qu'il est délicat d'effectuer car cela réalise déjà une interprétation de la tâche que l'utilisateur réalisera et donc une vue restreinte du contenu des emails. Dans l'exemple précédent (figure 9) aucun nettoyage n'a été réalisé.

**Digest de différence** Le digest affiche la différence entre deux ou plusieurs emails.

Le digest de différence permet de connaître les modifications effectuées entre deux emails (figure 10). Les nouvelles données qui apparaissent sont indiquées en jaune, alors que les anciennes sont rayées et indiquées en rouge. Ce type de digest permet de connaître les modifications faites par une tierce personne (révision d'un texte par exemple), ou alors si une information est mise à jour (date, lieu d'une réunion) alors la nouvelle information sera mise en avant.

IMPORTANT DATES (EXTENDED)

- Paper submissions : ~~March 20, 2011~~ April 4, 2011 (extended deadline)
- Paper notification : June 1, 2011
- Workshop : August 22, 2011 (full day)

Figure 10: Digest de différence.

**Digest géo-temporel.** Le digest affiche les dates et les lieux sur un support adapté (respectivement une ligne temporelle ou une carte géographique)

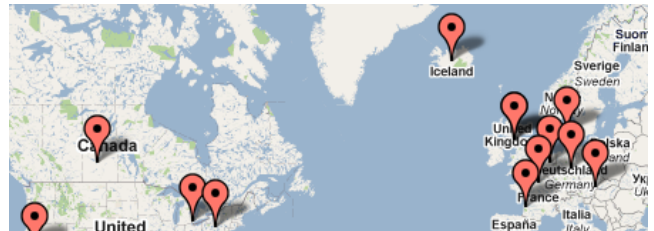


Figure 11: Digest géo-temporel, exemple d'une carte géographique.

Le digest géo-temporel (figure 11) permet une contextualisation simple et objective des données. Aucune tâche n'est extraite à partir de ces données.

**Digest de graphe de communication.** Le digest génère un graphe dont les noeuds sont les émetteurs et destinataires d'emails, et les arêtes les messages échangés.

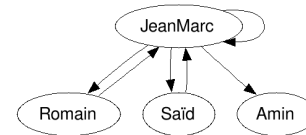


Figure 12: Graphe de communication.

Le digest de graphe (figure 12) ne considère pas les données contenues dans le corps emails, mais dans les en-têtes (expéditeurs et destinataire) qui sont déjà semi-structurés. Ce type de graphe, tout comme les nuages de mots, permettent l'analyse d'une grande quantité d'emails afin d'en extraire une information globale. Le graphe est généré par un appel au service web de Google Visualization [11] qui intègre la bibliothèque de visualisation de graphe GraphViz [7]. Le graphe généré est une image incluse dans l'email.

## 6. PROTOTYPAGE ET ITERATIONS

Un prototype de webmail a été implémenté en GWT<sup>1</sup> (*Google Web Toolkit*), ainsi qu'un add-on Thunderbird, implémentant la technique de digest d'emails. Ces deux prototypes font appel aux services web de traitement de données et de visualisation, et incluent en retour les digests sous

<sup>1</sup>Disponible à l'adresse suivante <http://digestme.appspot.com/>

forme d'emails. Ces prototypes ont permis de communiquer la technique de digest à la fois au sein de notre groupe de travail, ainsi que vers notre partenaire de tests (l'APCE, Agence Pour la Création d'Entreprises), partenaire du projet DLM 3.0.

Une étude informelle a été réalisée chez ce partenaire et nous a permis d'obtenir un retour avant le développement stable et la mise en production par la société Alinto. Une synthèse de ces retours est la suivante :

- Les utilisateurs demandent finalement peu de traitements automatiques. La nature cruciale des emails ne permet pas de passer à côté d'une information peut-être très importante.
- La navigation par légende latérale est pratique et pourrait être généralisée à tout type d'email trop difficile à parcourir avec une barre de navigation latérale qui devient trop petite si l'email est trop long.
- L'extraction d'entités est intéressante, mais les utilisateurs voudraient en sélectionner certaines en particulier (dates ou personnes importantes) qu'il faudrait mettre systématiquement en avant pour mieux les repérer au sein des emails.

Il sera par la suite nécessaire de réaliser des tests avec tâches et emails réels afin d'évaluer l'efficacité de cette technique sur une période de temps significative.

## 7. CONTRAINTES DE MISE EN PRODUCTION

La mise en production consiste à rendre les digests disponibles pour des utilisateurs d'un webmail existant (celui de la société Alinto). Nous décrivons deux principales contraintes liées à cette mise en production : les contraintes de performance et d'ergonomie.

### 7.1 Contrainte de performances

Cette contrainte est liée à deux paramètres. D'une part au nombre de messages traités et d'autre part au nombre d'utilisateurs accédant au service. Le service doit donc répondre suffisamment vite à l'utilisateur qui en fait la demande, tout en restant disponible aux autres utilisateurs sans latence excessive.

La réactivité des digests, à savoir leur temps de génération et de transfert, est essentiel pour leur adoption par des utilisateurs. Ce paramètre est dépendant de plusieurs intermédiaires : récupération des messages sur le serveur IMAP, transfert au service web pour calcul, et transfert retour pour affichage. Si dans le cas d'un client lourd on suppose que les messages sont disponibles en local (sur la machine de l'utilisateur), économisant ainsi une collecte sur le serveur IMAP, la problématique reste valable pour le temps de transfert du client vers le service web et pour le temps de calcul nécessaire au service web (le retour est probablement plus léger, puisque présentant une synthèse). Les axes d'optimisation du temps de réponse sont les suivants :

1. Pré-calcul à l'arrivée des messages (extraction de lieux, de noms, tokenisation).
2. Limitation de la fonctionnalité à la INBOX (indisponible dans les autres dossiers).

3. Définition d'un maximum de message pour un digest (le maximum peut varier selon le mode de digest choisi).

## 7.2 Contraintes ergonomiques

Un digest, en tant que message agrégé, possédant les caractéristiques d'un nouveau message (possibilité d'imprimer, de répondre, de transférer, d'afficher le source...), est un concept nouveau pour l'utilisateur. Cette nouveauté peut le déstabiliser de prime abord. La présentation du digest doit lui permettre de saisir à la fois la nature de message du digest et son origine (agrégation de plusieurs messages).

Un autre point est de conserver la cohérence de l'application. Un digest étant considéré comme un message à part entière, l'application doit proposer le même panel d'opérations pour un digest que pour un message simple. Ici, deux attitudes sont possibles : soit le digest est enregistré comme nouveau message dans la boîte IMAP de l'utilisateur, soit le digest est simplement affiché, sans existence persistante. Le deuxième cas implique davantage d'adaptations. Certaines des opérations courantes peuvent être portées naturellement : imprimer, transférer. D'autres doivent être désactivées : un digest simplement affiché ne peut pas être supprimé, ni marqué comme lu ou non lu ; le marquer comme spam n'a pas de sens non plus. Enfin, certaines opérations doivent adopter un comportement spécifique, comme le bouton de réponse : doit-on répondre à tous les expéditeurs de tous les messages constituant le digest ? Les questions de cet ordre sont nombreuses et, si elles ne doivent pas décourager la nouveauté, méritent qu'on s'y attarde afin d'apporter des réponses efficaces, pertinentes et suffisamment intuitives.

Finalement, la question de la sécurité doit être abordée. Avec une fonctionnalité d'agrégation en digest, les problématiques classiques de vie privée, d'envoi à la mauvaise personne, sont multipliées. Par exemple, l'opération "Répondre à tous" change subtilement de portée : il ne s'agit plus de répondre à tous les destinataires initiaux d'un message, mais à tous les expéditeurs des messages du digests. La nuance est de taille : ces contacts n'ont potentiellement échangé aucun message entre eux, et peuvent ne pas souhaiter que leur correspondance soit rediffusée à d'autres. De ce point de vue, l'opération "répondre à tous" appliquée à un digest se rapproche davantage d'une opération "Transmettre" préremplie que d'une réponse. Comme expliqué plus haut, le digest doit être présenté de la façon la plus compréhensible possible afin de rendre l'utilisateur parfaitement conscient de ses actes.

L'autre point lié à la vie privée est, comme pour toute analyse sémantique, l'envoi des messages à un service d'analyse externe. Cette transmission automatique du message peut être problématique pour un utilisateur ou une institution, autant expéditeurs que destinataires des messages concernés.

## 8. DISCUSSIONS ET CONCLUSION

Suite aux premiers tests sur le prototype et aux identification des contraintes de mise en production, nous avons itéré sur notre première version en rajoutant de nouvelles fonctionnalités. Nous discutons également dans quelle mesure les digests permettent la collaboration visuelle, ainsi que les limites actuelles de cette technique.

### 8.1 Itération

Un nouveau digest de coloration d'entités (figure 13) a été introduit afin de répondre aux besoins des utilisateurs.

Ainsi, certains éléments deviennent saillants car colorés dans les emails (figure 13). Ces éléments à colorier sont définis par les utilisateurs, via une interface externe (choix dans un dictionnaire). L'ajout de cette interface externe rompt avec notre principe de garder une interface unique orientée mail. Mais elle est nécessaire afin de permettre à l'utilisateur de gérer son PIM et les informations qu'il considère comme importantes.

*Digest de coloration d'entité.* Le digest colore les entités présentes dans un dictionnaire et que l'utilisateur souhaite voir mettre en avant dans un email.

NextMail'11  
 First International Workshop on Ne  
 August 22, 2011 - Lyon, France

PROGRAM COMMITTEE (IN PRC  
 - David Ascher CEO, Mozilla Messag  
 - Vitor R. Carvalho, CMU, USA  
 - Andrew Lampert, CSIRO, Australia

Figure 13: Coloration d'entités telles que des dates, lieux et personnes.

## 8.2 Analyse visuelle collaborative

Les digests pouvant être annotés, transférés ou archivés (figure 14), ils permettent la collaboration visuelle interactive. Les systèmes existants, tels que ManyEyes [27], sont souvent liés à un cadre technique (Java en l'occurrence) qui ne permettent pas l'export facile des données, aussi bien concernant la visualisation que les données sources. Les digests sont une solution permettant à l'utilisateur de conserver ses propres données (les emails) sans avoir à les rendre publiques ou sans avoir à les transférer sur un serveur distant. Même si les emails ne possèdent pas d'URL visible à l'échelle du web, ils peuvent cependant être identifiés (au moyen de leur champs d'en-tête `message-id`) comme des ressources uniques [12]. L'implémentation en services web permet à tout message ou toute visualisation générée d'être accessible sous forme de page web, et ainsi être partagé publiquement à plus grande échelle. Les digests respectant les standards des emails, en particulier au niveau de la mise en forme du contenu, ils pourront être lus sur tout type de client email ou navigateur web.

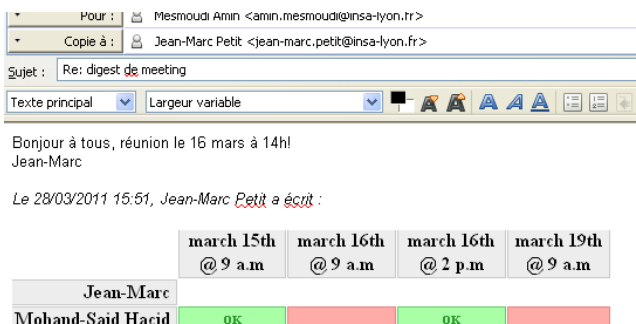


Figure 14: Transfert d'un digest comme un email normal.

## 8.3 Limites

Une première limite est propre à l'hypothèse initiale faite, à savoir de rester dans un cadre technique existant centré sur l'email. Les digests sont des vues relativement statiques car les seuls paramètres que peuvent faire varier l'utilisateur sont 1) le choix des emails et 2) le choix des digests. L'utilisateur se trouve ainsi limité s'il souhaite explorer l'espace de données et l'espace de design des visualisation incluses dans les digests. Une perspective est de permettre une interface externe de création de digests permettent aux utilisateurs de facilement créer un digest ou d'en modifier un. Une autre perspective, comme indiqué dans le paragraphe précédent, est de permettre le partage à l'échelle du web des visualisations et d'assister (sur le web) l'exploration des données et de l'espace de design (avec des légendes interactives par exemple).

Une autre limite est liée aux techniques de traitement de la langue (TAL). Ces techniques sont relativement matures sur le repérage d'entités nommées (cf. OpenCalais, OpenNLP, NLTK, ...), mais restent à l'état de prototype sur la détection libre d'événements complexes. (cf. les travaux de Xerox ou du CEA-List). A cette complexité intrinsèque liée à l'état de l'art des technologies du TAL, s'ajoute une seconde difficulté liée au contexte applicatif : le fait que l'utilisateur soit libre de sélectionner n'importe quel ensemble d'e-mails pour demander n'importe quel type d'"email digest" a pour conséquence une explosion des possibilités qui entrainera nécessairement une dégradation de la qualité du repérage. En effet, ces grammaires ont une précision d'autant plus forte que l'objet de la recherche est restreint et connu. Cette limitation (bloquante pour le passage à l'échelle) pourrait être contournée par le choix de certains cas d'utilisation restreints, par exemple l'organisation de rendez-vous ou bien le repérage des lieux pour créer un digest géographique ou encore l'analyse d'un dossier de mail d'alertes : si l'objet de la recherche est restreint et connu, la qualité de reconnaissance pourra être suffisante, alors qu'une détection "à l'aveugle" conduirait probablement à un fort silence, décevant pour l'utilisateur. En d'autres termes, restreindre le type de données en entrée ainsi que la nature des objets recherchés est une condition nécessaire à la mise en oeuvre de telles techniques.

## 9. REMERCIEMENTS

Ces travaux ont été partiellement financés par le projet DLM 3.0 (<http://www.dlm30.com/>) dédiés à l'email sémantique et soutenu par le ministère de l'économie et des finances dans le cadre du volet numérique du plan de relance 2009.

## 10. REFERENCES

- [1] Bacn. <http://en.wikipedia.org/wiki/Bacn>. 2011.
- [2] R. Beverly. A Human Factors Approach to Spam Filtering. In *Conference on Email and Anti-Spam*, 2009.
- [3] G. Carenini, R. T. Ng, and X. Zhou. Summarizing email conversations with clue words. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 91–100, New York, NY, USA, 2007. ACM.
- [4] F. Chevalier, S. Huot, and J. Fekete. WikipediaViz: Conveying article quality for casual Wikipedia readers.

- In *Pacific Visualization Symposium (PacificVis), 2010 IEEE*, pages 49–56. IEEE, 2010.
- [5] Doodle. <http://www.doodle.com/>. 2011.
- [6] N. Ducheneaut and V. Bellotti. E-mail as habitat: an exploration of embedded personal information management. *interactions*, 8:30–38, September 2001.
- [7] J. Ellson, E. Gansner, L. Koutsofios, S. North, and G. Woodhull. Graphviz – open source graph drawing tools. In *Graph Drawing*, pages 594–597. Springer, 2002.
- [8] O. Etzioni, M. Banko, S. Soderland, and D. S. Weld. Open information extraction from the web. *Commun. ACM*, 51:68–74, December 2008.
- [9] D. Gibson, K. Punera, and A. Tomkins. The volume and evolution of web page templates. In *Special interest tracks and posters of the 14th international conference on World Wide Web, WWW '05*, pages 830–839, New York, NY, USA, 2005. ACM.
- [10] gmail. <http://www.gmail.com/>. 2011.
- [11] google visualization. <http://code.google.com/apis/chart/interactive/>. 2011.
- [12] J. Heer, F. Ham, S. Carpendale, C. Weaver, and P. Isenberg. Creation and collaboration: Engaging new audiences for information visualization. pages 92–133, 2008.
- [13] J. Jacko and A. Sears. *The human-computer interaction handbook: fundamentals, evolving technologies, and emerging applications*. CRC Press, 2003.
- [14] W. Jones. Personal information management. *Annual review of information science and technology*, 41(1):453–504, 2007.
- [15] H. Kang, C. Plaisant, T. Elsayed, and D. W. Oard. Making sense of archived e-mail: Exploring the enron collection with netlens. *J. Am. Soc. Inf. Sci. Technol.*, 61:723–744, April 2010.
- [16] R. Kumar, J. Talton, S. Ahmad, and S. Klemmer. Bricolage: Example-Based Retargeting for Web Design.
- [17] O. Lassila and R. Swick. Resource description framework (rdf) model and syntax. *World Wide Web Consortium*, <http://www.w3.org/TR/WD-rdf-syntax>.
- [18] W. Mackay. More than just a communication system: diversity in the use of electronic mail. In *Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, pages 344–353. ACM, 1988 .
- [19] D. Oswald, S. Raha, and I. Macfarlane. HTML Parser. *SourceForge. net*.
- [20] T. Parr. Enforcing strict model-view separation in template engines. In *Proceedings of the 13th international conference on World Wide Web*, pages 224–233. ACM, 2004.
- [21] Quote Colors. <http://quotecolors.mozdev.org/>. 2011.
- [22] M. Skeels, B. Lee, G. Smith, and G. Robertson. Revealing uncertainty for information visualization. *Information Visualization*, 9(1):70–81, 2009.
- [23] Y. Tian, R. A. Hankins, and J. M. Patel. Efficient aggregation for graph summarization. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data, SIGMOD '08*, pages 567–580, New York, NY, USA, 2008. ACM.
- [24] F. Viégas, S. Golder, and J. Donath. Visualizing email content: portraying relationships from conversational histories. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 979–988. ACM, 2006.
- [25] R. Vuillemot, J.-M. Petit, and M.-S. Hacid. Shift-BOX: INBOX Time Shifting to Reduce Email Clutter. In *Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, July 2010.
- [26] R. Vuillemot, J.-M. Petit, and M.-S. Hacid. Generalizing Email Messages Digests. In U. ACM New York, NY, editor, *CHI 2011 - 29th ACM Conference on Human Factors in Computing Systems (Extended Abstract)*, May 2011.
- [27] M. Wattenberg, J. Kriss, and M. McKeon. Manyeyes: a site for visualization at internet scale. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1121–1128, 2007. Member-Viegas, Fernanda B. and Member-van Ham, Frank.
- [28] A. Watters. How Recent Changes to Twitter’s Terms of Service Might Hurt Academic Research. 2011.
- [29] S. Whittaker and C. Sidner. Email overload: exploring personal information management of email. In *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*, pages 276–283. ACM, 1996.