BlueSpace Mail Relevance

Contents

1.1 Index

1.1 INDEX 2
1.2 LIST OF FIGURES 3
1.3 LIST OF TABLES 4

2 EXECUTIVE SUMMARY 5

3 INTRODUCTION 6

3.1 EMAIL 6
3.2 EMAIL SEARCH 6
3.3 BLUESPACE 6
3.4 BUSINESS BENEFITS 6
3.5 USER IMPACT 7

4 BLUESPACE TECHNOLOGY 8

4.1 BLUESPACE RELATIONAL ARCHITECTURE 8
4.2 TRANSMAIL AND SUBMESSAGE PATHS 8
4.3 BLUESPACE EXTERNAL TRANSMAIL 10
4.4 BLUESPACE GOVERNANCE 11

5 BLUESPACE TRANSMAIL RELEVANCE 13

5.1 INTRODUCTION 13
5.2 OVERVIEW 14
5.3 MAIL DATA 17
5.3.1 Mail data inputs 17
5.3.2 Mail data relevance 18
5.3.3 Mail data collaborative relevance 19
5.4 MAIL ACTIONS 19
5.4.1 Mail actions inputs 19
5.4.2 Mail actions relevance 20
5.4.3 Mail actions collaborative relevance 21
5.5 MAIL ACTIONS IMPORTANCE 22
5.6 CORRELATED ENTITIES 23
5.7 ROLES 23
5.8 SIMILARITY 24
5.9 EXTENSIONS 24
5.10 USER INTERACTION 24
5.11 SUMMARY 25

6 EMAIL COMPARISON 26

6.1 INTRODUCTION 26
6.2 EMAIL UNIT 26
6.3 RELATIONAL RESTRUCTURING 26
6.4 EMAIL INFRASTRUCTURE INTEGRATION 27
6.5 FORENSICS 28
6.6 SUMMARY 29

7 USER INTERFACE 30

7.1 INTRODUCTION 30
7.2 BLUESPACE TRANSMAIL VIEW 30
7.2.1 Mail view 30
7.2.2 Folder view 36
7.2.3 Address book 36
7.2.4 Preferences 37

7.3 BLUESPACE TRANSMAIL SEARCH 39
7.3.1 Search form 39
7.3.2 Advanced search 41
7.3.3 Custom phrases 46
7.3.4 Search settings 46
7.4 BLUESPACE TRANSMAIL SEARCH RESULTS 48
7.4.1 Search results 48
7.4.2 Folder view 48
7.4.3 Extended view 50
7.5 SUMMARY 53

8 BLOGS 54

8.1 INTRODUCTION 54
8.2 RELEVANCE IN BLOGS 54

9 MATHEMATICAL ALGORITHM FRAMEWORK 55

9.1 INTRODUCTION 55
9.2 AIM OF RELEVANCE FORMULA 55
9.3 FACTORS IN RELEVANCE FORMULA 55
9.4 RELEVANCE FEEDBACK LOOP 58
9.5 IMPORTANCE 58
9.5.1 Factors affecting global importances over time 59
9.5.2 Factors affecting local Importance over time 60
9.6 CONTENT ANALYSIS 62
9.7 POLICING THE SYSTEM 63
9.8 THE POSSIBILITY OF SERIOUS ERROR 63
9.9 SUMMARY 63

10 SOFTWARE IMPLEMENTATION 64

10.1 INTRODUCTION 64
10.2 LOCAL AND GLOBAL VALUES 69
10.3 SIMILARITY 74
10.3.1 Role or group based 75
10.3.2 Collaborative filtering 75
10.3.3 Fingerprinting 76
10.3.4 Cluster models 76
10.3.5 Search-based methods 77
10.3.6 Frequency analysis 77
10.3.7 Combinations and detail 77
10.3.8 Summary 79

11 CONCLUSION 82

12 GLOSSARY 83

1.2 List of Figures

Figure 1 – Email versus TransMail 8
Figure 2 – BlueSpace TransMail scenario 9
Figure 3 – Branching TransMail paths 10
Figure 4 – TransMail to external contacts 11
Figure 5 – Complete compliance model 12
Figure 6 – Relevance overview 15
Figure 7 – BlueSpace email integration 27
Figure 8 – BlueSpace TransMail view 31
Figure 9 – Asking the user to rate content relevance 33
Figure 10 – BlueSpace TransMail user view 34

Figure 11 – Importance evolution *35*
Figure 12 – BlueSpace TransMail folder view *36*
Figure 13 – BlueSpace address book *37*
Figure 14 – Tuning the algorithm *38*
Figure 15 – Simple search interface *40*
Figure 16 – Regular search interface *41*
Figure 17 – Advanced search interface *42*
Figure 18 – Search settings *47*
Figure 19 – Folder view search results *49*
Figure 20 – Extended view search results *50*

1.3 List of Tables

Table 1 – TransMail view icons *32*
Table 2 – Message details icons *35*
Table 3 – Algorithm tuning icons *38*
Table 4 – Advanced search icons *43*
Table 5 – Search settings icons *47*
Table 6 – Folder view icons *49*
Table 7 – Extended view icons *51*

2 Executive Summary

E-mail was invented in the early 1970s as part of an academic experiment.[i] According to META Group, 75% of corporate knowledge will reside in email by 2005.[ii] Today, business runs on email – it gets used for the vast majority of interactions between people. The volume of email is now colossal, with many users receiving in excess of 200 emails a day.

The challenge for employees is to navigate through this 'information torrent'. To discover what is actually relevant to them through a sea of noise. This problem is particularly acute for management employees.

Due to the volume of email, it is often the primary source of evidence in legal, regulatory and intelligence investigations. This volume has forced sampling of email content (as opposed to manual reading of all content) in investigative processes. Any techniques that can improve the sampling process, or provide better high-level representations of volumes of content, can assist forensics.

BlueSpace employs a relational architecture to add structure to mail content. Each piece of content is audited and managed through its lifecycle in the mail system, and every significant user action is audited. This provides a powerful set of raw data that can be used in relevance algorithms.

Most email systems today have basic search facilities in which each email is given equal weighting. BlueSpace allows searching by keyword, process code and other such criteria. But in addition, BlueSpace uses a multi-dimensional relevance system – BlueSpace Mail Relevance – which rates the importance of each piece of content based primarily on the following criteria:

1. **User actions** – what did the user do with the mail? If they moved it to a folder or forwarded it then it is likely to be of significant importance.
2. **User collaborative actions** – what did other users do with the mail? Again, their actions indicate their views on the content's importance.
3. **Administrative settings** – a 'Relevance Administrator' can add weightings reflecting the enterprise's view of what is important.

BlueSpace Mail Relevance does not only rate content – it also rates the importance of each user and other entity (such as Jobcodes and contexts) in the system.

BlueSpace Mail Relevance provides much needed innovation in the user experience of electronic mail, such as helping them handle the most important mail content first. The search facilities are far in advance of any comparable system. A user can change the relevance algorithm in real-time to improve search results – e.g. "show me content that was read by more people". A user can also view what happened to mail content they have written during its lifecycle, including recipients, read times and the path the content took. The relevance capabilities can also greatly assist regulatory and intelligence agencies in improving their forensics processes on regular email content.

Note that both users and administrators do not need to interact with the BlueSpace search algorithm (e.g. changing relevance/importance) for the system to work. It functions automatically based on what users do – though users can provide feedback to help the algorithm learn.

Email has become both the largest unstructured information store and the most comprehensive repository of knowledge in modern corporations. Enterprises and their employees need more effective mechanisms for mining the information in their mail systems. BlueSpace Mail Relevance provides such a mechanism. The granularity of the BlueSpace architecture allows access to the raw data required to deploy a highly relevant search system. BlueSpace also allows people to find the content they're looking for in email.

3 Introduction

3.1 Email

Email was invented in the early 1970s as part of an academic experiment.[iii] While its fundamental protocols (such as Simple Mail Transfer Protocol – SMTP) have changed little since their inception in 1982,[iv] real-world usage of email has changed dramatically. Email is now a mission-critical business tool, and one of the most resource-hungry services provided by IT. Analyst firm Gartner has indicated that the volume of email is growing by 50% every year,[v] causing companies' bandwidth requirements to double every six months.[vi]

SMTP treats each email as a string of text similar to a postcard or telegram that must be passed between computers until it reaches its desired recipients. When a user replies to or forwards on an email, they photocopy all the content in that email and put it into a new email text string with their own comments. This has made email systems into a 'content dumpster', with no structure or ability to audit content, or to mine the information inside the email system.

According to META Group, 75% of corporate knowledge will reside in email by 2005.[vii] But there is no way to mine this data effectively. This is particularly important for email that constitutes formal business records. Most corporations allow users to view only the mail content they have received directly, whilst traditional content management systems have access rights layered over the entire content store. This again reflects the history of email as a person-to-person communications system, rather than a system designed for knowledge management.

3.2 Email Search

Email search is a disparate market, with different products offering varying value propositions. Some companies offer a query-like store as a way to remove content from an active mail server to help such servers scale. Others have attempted to branch into email search from content management systems, but have faced challenges due to the lack of structure in the mail system and the lack of defined access rights. If a corporation would just allow everyone access to all emails, it would be carnage.

3.3 BlueSpace

BlueSpace uses its unique relational technology to provide greater granularity into content in a mail system. By coupling this to its governance model, BlueSpace can determine which pieces of a content a user should be allowed to search. The innovation described in this white paper involves adding advanced analytics into a search system in BlueSpace, causing a dramatic improvement in relevance for the end user attempting to find content.

This does not only apply to search, however, but also helps the user to get better at 'email browsing'. Many users in corporations are now at the sensible limits of information exposure through email, receiving up to 500 emails each day. BlueSpace email relevance is designed to help deal with this problem.

3.4 Business Benefits

The concepts and technology described in this white paper will benefit all businesses and organizations to some level, but there is one category for which it may become transformational – research. Organizations that have a

focus on research, particularly collaborative research, have many systems that attempt to leverage knowledge at the enterprise level by encouraging information sharing between individuals. But the user community tends to gravitate towards email, and rarely complies with rigorous document management policies due to the sheer volume of email content

Specific industry examples include:

- Pharmaceutical firms, in which researchers have focused scientific specialities, but need to collaborate in cross-specialist teams.
- Financial research functions, in which analysts produce reports for customers who subscribe to their distribution lists.
- Intelligence agencies, that generate large volumes of intelligence data, and then need to distribute this data to 'customers' while also rating its likelihood of truth or some other criteria based on real-time user corroboration.
- Investigatory agencies, such as a tax inspection service, who investigate individuals and organizations for malpractice, and desire better ways to find the 'needles in the haystack' through correlative inspection.

More general benefits and uses for the concepts and technology described in this white paper include:

- Recognizing Spam and decreasing the volume of users who waste time dealing with it – this applies to both external 'advertising' mail and internal 'noise' mail.
- Discouraging dysfunctional mail behaviour by end users, in which people unnecessarily copy many recipients on mails they write.
- Facilitating shared inbox efficiency (e.g. between managers and their secretaries).
- Allowing users to deal with the most important content first, e.g. when they come back from holiday and have 2000 mails in their inbox.
- For mobile users, only pushing the content by default that is judged to be the most relevant to them.
- Improving overall efficiency by helping users to better manage their mail 'information torrent' – a process that a typical user carries out for between two and five hours each day.

3.5 User Impact

Much of the innovation in previously published BlueSpace white papers has focused on either enterprise governance and control, or infrastructure efficiency or automation. An obvious question to ask is, "Will people use, adopt and 'fall in love with' the BlueSpace application?" This white paper outlines a powerful addition to the user experience – greatly improved and collaborative search capabilities that help the user navigate their mail. Email may have become the enterprises' core knowledge management repository. But it is ironic that the current, mainstream search facilities for end users in email applications such as Microsoft Outlook are so antiquated and ineffective.

4 BlueSpace Technology

4.1 BlueSpace Relational Architecture

BlueSpace has developed a messaging platform that is very different from existing messaging and mail systems. As introduced in Section 3.1, email photocopies content with each generation of reply and forward, as shown in Figure 1.

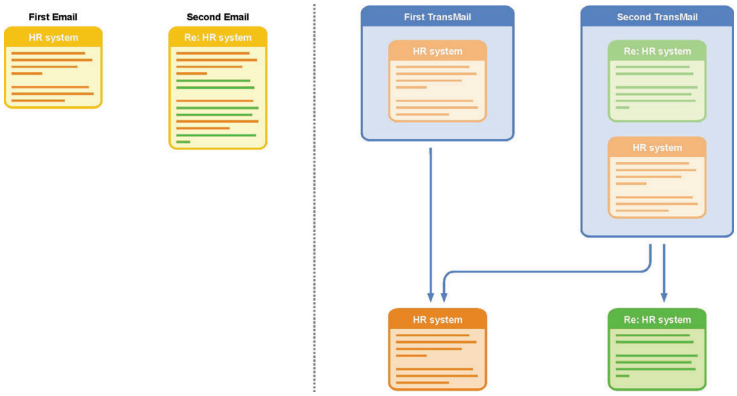

Figure 1 – Email versus TransMail

In the second email, the red message is duplicated without any reference to its origin – there is no link between the first email and the second email. In the BlueSpace TransMail system, the first TransMail points at the red message, and the second TransMail points at both the red and green messages.[viii] Both red and green messages are termed 'submessages' in the BlueSpace environment. There is a glossary defining all terminology in this white paper in Section 12.

This relational pointing has two powerful outcomes:

1. **Control:** It is far easier to manage and audit the content, enforce access rights and guarantee retention policies are rigorously applied, as there is just a single copy of the message to manage.
2. **Cost saving:** The TransMail system uses far less infrastructure than regular email systems, as it does not photocopy the same piece of content in to many separate places on the infrastructure.

4.2 TransMail And Submessage Paths

An important concept in the world of BlueSpace is that of paths – the way information is moved around a BlueSpace system. Due to the relational architecture of the BlueSpace TransMail application, BlueSpace is able to track what happens to content during its lifecycle, as well as enforcing rules over what is allowed to happen to it.

This section is included to assist the reader in understanding the relational model of BlueSpace, and its importance in search relevance. The user is not required or expected to go to this level of detail when using the application – it's irrelevant to them, they just send mail.

Figure 2 illustrates an example of a series of BlueSpace TransMail communications in which the user Justin Marston is involved.

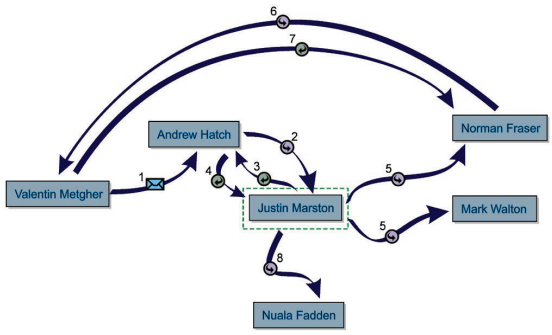

Figure 2 – BlueSpace TransMail scenario

The steps 1 to 8 in Figure 2 are as follows:

1. Valentin Metgher sends a TransMail to Andrew Hatch that contains submessage A.
2. Andrew Hatch forwards the TransMail to Justin Marston, creating a new TransMail that contains B|A (a new submessage B, with history portion submessage A).
3. Justin Marston replies to Andrew Hatch, creating a TransMail that contains C|BA.
4. Andrew Hatch replies to Justin Marston, creating a TransMail that contains D|CBA.
5. Justin Marston forwards the TransMail generated in step 4 to Norman Fraser and Mark Walton, creating a TransMail that contains E|DCBA.
6. Norman Fraser seeks to clarify the situation with Valentin Metgher, sending a TransMail containing F|EDCBA.
7. Valentin Metgher replies to Norman Fraser, creating a TransMail that contains G|FEDCBA.
8. Some time later, Justin Marston forwards the TransMail he received from Andrew Hatch to Nuala Fadden, creating a TransMail containing H|DCBA.

There are three types of submessage with respect to any given TransMail or other submessage:

1. **Current submessage:** The submessage at the top of the chain in any given TransMail is termed the ‘current submessage’, which will have been written by the user who created the TransMail. In Figure 1, the red submessage is the current submessage in the first TransMail, and the green submessage is the current submessage in the second TransMail. In step 5 of Figure 2, submessage E is the current submessage.
2. **History submessage:** The previous submessages in the chain of any given TransMail (e.g. the red submessage in the second TransMail in Figure 1) are termed ‘history submessages’. History submessages are content that was created before the current submessage and its respective TransMail. In step 5 of Figure 2, submessages D, C, B and A are the history submessages.

3. **Future submessage:** For any given submessage, submessages that appear in later TransMails that reference the original submessage are termed 'future submessages'. Thus in Figure 1, the green submessage is a future submessage with respect to the red submessage. In step 5 of Figure 2, submessages F and G are the future submessages, but not submessage H, as the TransMail generated in step 8 does not include submessage E. H is a future submessage in step 4 of Figure 2.

Note that a submessage can only have one history path, as it can be the 'current submessage' in only one TransMail. A submessage can, however, have many future paths, as it may be included in many branching forward and reply structures. This is illustrated in Figure 3.

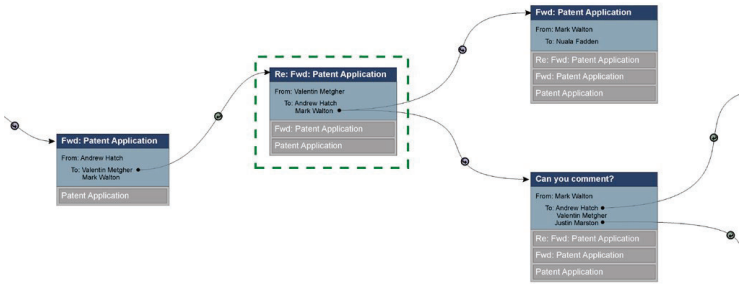

Figure 3 – Branching TransMail paths

All definitions of history and future are thus relative to a given point in time. In Figure 3, the ringed TransMail from Valentin Metgher to Andrew Hatch and Mark Walton is being considered as current. History submessages were written as part of the TransMails to the left, Future submessages were written as part of the TransMails to the right.

Note that a user may have already read content that constitutes both history and future submessages as part of other TransMails when reading a given TransMail.

4.3 BlueSpace External TransMail

BlueSpace can allow TransMail users to send content to external contacts, as shown in Figure 4.

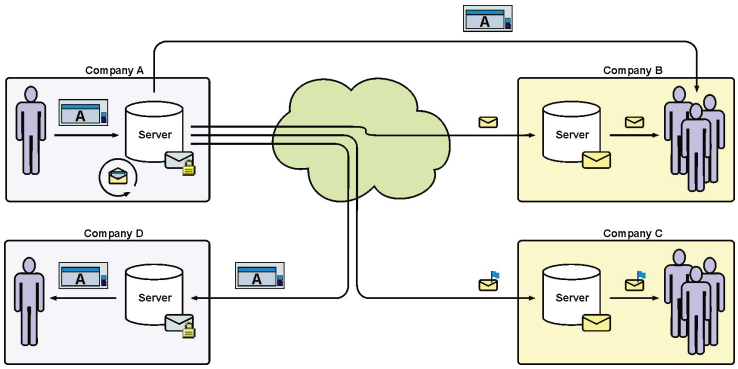

Figure 4 – TransMail to external contacts

Figure 4 illustrates three different methods:

1. A user in Company A sends a TransMail to users in Company B. The BlueSpace server in Company A sends an email notification (with a URL link to the TransMail) to the recipients in Company B. The recipients in Company B then click the URL to read the TransMail through a web browser.
2. A user in Company A sends a TransMail to users in Company C. The BlueSpace server in Company A wraps the TransMail up in an SMTP envelope, and sends it like a regular email to the email server in Company C. The recipients in Company C view it as normal. (Note that some control is lost by Company A, as it is unaware of the actions of users in Company C.)
3. A user in Company A sends a TransMail to a user in Company D. The BlueSpace server in Company A realises that the mail server in Company D is a BlueSpace server using BlueSpace Discovery Service.[ix] The BlueSpace server in Company A sends the TransMail to that in Company D using a protocol based on web services, and the recipient in Company D then reads the TransMail.

The second scenario, in which BlueSpace wraps a TransMail in an SMTP envelope, is the most complex to manage at the application level. Sending out the email is easy enough, but interpreting any replies can be a challenge, especially in corporations with BlueSpace systems deployed alongside regular email systems. This has been discussed in detail in the BlueSpace Email Structuring white paper.[x]

4.4 BlueSpace Governance

BlueSpace manages content during its entire lifecycle according to rules laid down in the BlueSpace Integrated Governance Management[xi] model. This determines how long content is kept for, who can access it, where the data should reside, and the other allowable actions to be carried out by users.

A UML representation of the complete BlueSpace compliance model is presented in Figure 5.

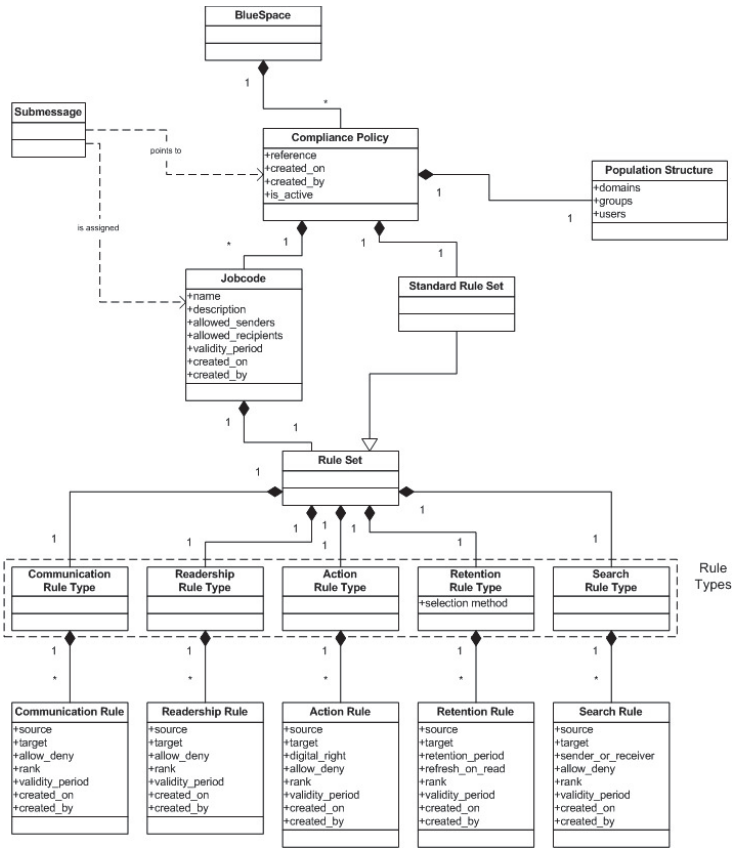

Figure 5 – Complete compliance model

The key aspect for this white paper is the ability of BlueSpace to layer search rules over a content system, determining who has the right to search different pieces of content in the system. For example, in an investment bank, an analyst might have the right to search all content related to analyst processes written by other analysts. However, the same analyst should not have the ability to search any content that does not pertain to his/her role in the bank.

The BlueSpace search rules technology is fully described in the BlueSpace Integrated Governance Management[xii] model.

5 BlueSpace TransMail Relevance

5.1 Introduction

BlueSpace TransMail Relevance is a complex concept, and to facilitate easier comprehension, this section provides a brief introduction to the principles involved.

Relevance has been designed to help users make sense of the vast quantities of mail they receive, by inferring a user's interest in content and other users based on monitoring that user's actions in the application. There are two primary elements to the BlueSpace TransMail Relevance model:

- **Relevance:** Applies to content, e.g. submessages.
- **Importance:** Applies to entities, e.g. users and Jobcodes.

It is highly important for the reader to understand each of these two terms, as understanding the remainder of this white paper depends on a firm grasp of these primary elements.

Relevance rates each piece of content in an order reflecting a user's likely interest in the content. A high relevance value indicates that the content is likely to be of significant interest to the user, while a low relevance value indicates that the content is likely to be of little interest to the user.

Importance rates each entity, such as each Jobcode, reflecting a user's likely interest in content involving that entity. A high importance value indicates that a given Jobcode is likely to be of significant interest to the user in question, while a low importance value indicates that a Jobcode is likely to be of little interest to the user.

But how do relevance and importance values get determined? The primary input for determining both relevance and importance is observing users' actions – an automatic, background process. Users' actions can indicate a lot about both content and entities in the system, for example:

- If a user receives a TransMail, saves it in a custom folder, and then forwards it to three other users, then it is likely to be much more relevant to them than if they had just deleted it as soon as they had finished reading it.
- It is likely that a user's line manager will be important to them, because they will typically read content from their line manager in a timely manner, will often reply to it, will send that line manager a significant number of TransMails, etc.

These may sound like arbitrary actions from which it is difficult to draw conclusions from any one given user action, but with the power of statistics and a very large sample size, the accuracy of the algorithm can be impressive. A typical user may receive between one hundred and two hundred electronic mails each day.

Another factor working in favour of BlueSpace TransMail Relevance is the collaborative aspect – actions of different users can be compared. This increases the sample size to potentially millions of electronic mails every day – a huge set of data on which to base judgements on relevance and importance.

To summarize, BlueSpace TransMail Relevance observes user's actions to infer their interest in both content and other entities (such as users) in a BlueSpace system.

5.2 Overview

As described in Section 4.1, BlueSpace separates each TransMail into its constituent message objects, written by different authors. This provides far greater granularity than any other messaging system. The granularity and ease of access to the information helps to support the powerful analytics that BlueSpace can apply to a TransMail system. Behavioural information can be easily gleaned from the mail environment and used to help improve content relevance.

BlueSpace TransMail Relevance helps users as they perform two different actions:

1. **Browsing:** In this mode, the user moves through a folder of email content, such as looking through their new TransMails in their inbox.
2. **Searching:** In this mode, the user is actively looking for a specific piece of information, or for information of a given type.

Note that relevance scores for a given piece of content are absolute when browsing (as there are no specific criteria), but vary when a user is searching (relative relevance). For example, a piece of content is more relevant to a search if two of the words in the subject are keyword search terms than if there is only one. The user interface only ever shows the user absolute relevance values – relative relevance values are used by the application to determine initial ranking of TransMails in search results.

There are two types of information that can be recovered from the mail system:

1. **Mail data:** Data in the mail itself, including keywords, creation dates, recipient information, Jobcodes and other meta-type data.
2. **Mail actions:** Information regarding users' actions on mail content, including the number of times it has been read, the number of times it has been forwarded, etc.

There is a glossary defining all terminology in this white paper in Section 12. In addition, end users can themselves rate various aspects and entities in the system to improve the algorithm's accuracy. They can provide direct feedback on the relevance of any given piece of content, or the importance of any given entity. They can also change various weighting factors that determine the settings of the algorithm, which are otherwise self-adjusting.

Users interact with relevance in three different ways:

1. Reading new mail – when a user receives a new TransMail, relevance indicates how useful that content is likely to be.
2. Reading existing mail – when a user is viewing their folder listing or a specific TransMail in the mail view, relevance indicates how important each TransMail and submessage is likely to be.
3. Searching mail – when a user is searching through TransMail to find something, relevance can help to rank the search results, and indicate to the user how likely each piece of content is to be of interest.

An 'importance' factor is determined for each entity (users, groups, Jobcodes etc) in the system, acting to weight various inputs to the relevance algorithms. Importance values are derived from historical learning in the system, based on previous actions by users.

A high-level illustration of the various elements and relationships discussed in this white paper is presented in Figure 6. This diagram is extremely useful in understanding the concepts involved in BlueSpace TransMail Relevance, and the reader is recommended to return to this diagram frequently during their reading of this white paper.

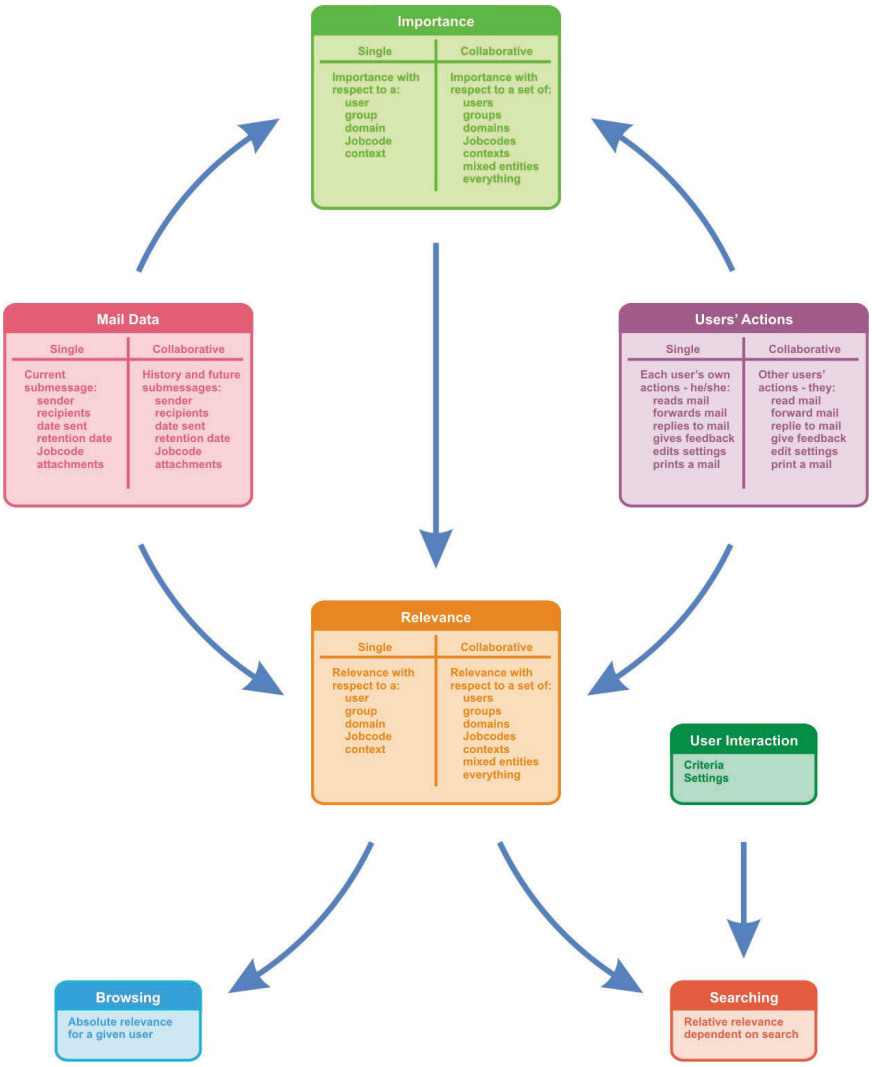

Figure 6 – Relevance overview

Figure 6 is clearly quite complex, but sums up the overall information flow in the BlueSpace relevance engine. 'Single' aspects refer to individual units (be they content, users, etc). 'Collaborative' aspects refer to collections of units (again be they content, users, etc). Each box in the diagram is described below:

- **Mail Data:** Mail data is properties of each mail, including the current submessage (single), as well as both history and future submessages (collaborative) – as defined in Section 4.2. Mail data acts as an input to importance, in that it determines the entities listed in the mail environment and where they appear. Mail data acts as an input to relevance, because the entities involved in the mail help to determine its initial and ongoing relevance values.

- **Users' Actions:** Users' actions are the ways users interact with the BlueSpace system, e.g. reading mails, replying to mails and changing their settings. Users' actions encompass a given user's own actions (a single user) and other users' actions (collaborative users in the system). These actions help to determine relevance, as relevance is calculated based on the way users interact with content. Users' actions also help to determine importance, as the importance of entities reflects the significance with which users treat them.
- **Importance:** The importance of a given entity can be stated with respect to a single entity (e.g. "how important is John to Douglas?") or a set of entities (e.g. "how important is John to the Calucite drug development team?"). Importance draws input from mail data, which determines the 'fields of engagement' for entities, and from users' actions, which reflect how significant a given entity is to a given user or set of users – 'the way the battle plays out'. Importance also acts as an input to relevance, as it helps to determine the relative weightings given to different actors in the system. Note that importance values for different entities are shown in a user's graphical interface (as described in Section 7).
- **Relevance:** The relevance of a given piece of content can be stated with respect to a single entity (e.g. "how relevant is it to John?") or with respect to a set of entities (e.g. "how relevant is it to the Calucite drug development team?"). Relevance has as inputs the mail data (properties of the content in question), users' actions (what users have done with the content), and importance (the significance users attribute to the entities involved in the content). Relevance outputs to two user activities – browsing (absolute relevance – helping users navigate their mail folders) and searching (relative relevance – determining the initial ranking of content for a specific search). Absolute relevance values are displayed for content – relative relevance values are never directly shown to end users in the interface. Users would get extremely confused by two different sets of relevance values!
- **User Interaction:** When users are searching for something, they have some idea of what they are looking for, and can guide the system to that content by stating criteria and changing their search settings.
- **Browsing:** This is the activity users spend most of their time doing in mail systems – navigating through mail folders and reading content. Relevance acts as an input here, helping to guide the user to the content they might wish to do deal with first. There is only one absolute relevance value for each piece of content with respect to a given user.
- **Searching:** This is the activity that often most frustrates end users – not being able to find the mails they are looking for amongst the thousands of mails they have sent and received. The relative relevance of content to a given search varies depending on the search criteria settings (and the user's settings).

There can be no doubt that this data schema is complex, as reflected by the mathematics and computing in Sections 9 and 10. It may appear to be circular – and it is – because the relevance system in BlueSpace employs a feedback loop. It does have access to relatively independent data points, however, both by indirect 'watching' (users' actions and mail data) and by direct 'questioning' (asking users whether the results are accurate). These are only relatively independent, however, as users' opinions of the significance of both entities and content will be influenced by the values presented to them by the relevance system, and implicitly their trust in this system based on their views of its accuracy.

The concepts described above may seem rather abstract, so the following sections of this white paper outline 'real-world' examples that help to illustrate the way relevance and importance work in practice.

5.3 Mail Data

This section is divided into three:

1. Mail data inputs – both conventional and BlueSpace-specific criteria that can be used to search content.
2. Mail data relevance – scenarios illustrating how this data can affect search relevance.
3. Mail data collaborative relevance – scenarios involving other pieces of content in determining the relevance of a given piece of content.

5.3.1 Mail data inputs

Many of the criteria for searching mail data in BlueSpace are the same as those in traditional mail search environments:

- Keywords in subject
- Keywords in message body
- Keywords in other fields
- Keywords in attachments (both names and content)
- Any signature assigned to the message
- The date the content was created
- The recipients of the content
- Whether it was sent To:/Cc:/Bcc:
- The sender of the content
- The body size
- The attachment size
- The attachment type
- Flags for follow up – the sender indicates the TransMail requires action by the recipient.

Note that a user may build a search query looking for different submessages that make up a TransMail, and may specify different creation dates, senders, recipients and other details for each submessage they are looking for that make up a TransMail. The user interface supporting these highly specific queries is detailed in Section 7.3.2.

Other criteria are unique to BlueSpace, and leverage the relational architecture of the BlueSpace TransMail system coupled to the strong emphasis BlueSpace places on business process integration:

- Jobcodes – a mechanism for linking an entire submessage (as defined in Section 4.1) to a specific business process.
- Context – a mechanism for linking any text in a submessage to a specific process, either on a user or enterprise-wide basis, as shall be discussed in the BlueSpace Contexts white paper.
- Groups – sets of users across the system.
- Domains – an X.400-style naming directory across the system.

- Priority – an urgency tag linked to notification processes.
- Sensitivity – a confidentiality tag linked to readership and search rights.
- Validity date – the period of time for which the content is viewable by the user.
- Retention date – the period of time for which the content is retained by the corporation.
- Allowed readers – the users who are allowed to read the mail, as determined by the governance rules.
- Allowed geographies – the geographies to which the mail content is allowed to be sent, as determined by the governance rules.
- Rule contravention – content sent that had weak conflicts (e.g. a history submessage could not be included for one of twenty recipients) or strong conflicts (e.g. a user tries to send a TransMail to another user with whom they are not allowed to communicate) with the governance rules, and was altered or managed appropriately by the application.
- Other governance interactions – various other interactions with the governance engine.
- Other Meta-style information stored relationally with respect to the mail.
- Other business processes that are triggered by the mail event.

5.3.2 Mail data relevance

Examples of ways in which these parameters might affect relevance to a given user, either while browsing or as part of a search, include:

- If the keyword they are looking for is in the body, then it's likely to be relevant to them.
- If a user is looking for a Microsoft Word file, then TransMails with word attachments are relevant to the search.
- If a user is looking for a piece of content they received some time ago from the HR department, which was pretty important, then all of those criteria (date period, priority and the HR group/domain) will be relevant to the search.
- If a user is looking for all content relating to a particular M&A transaction, and that content has been tagged with Jobcodes, then content with the appropriate Jobcode number will be relevant to the search.
- If a user is looking for content that may have caused compliance violations, then all content that caused weak or strong interactions with the governance rules will be relevant to the search.
- If a user received a TransMail as one of fifty other Cc: recipients, it is less likely to be of relevance to them than if they received the TransMail as the sole recipient.
- If a user is looking for all TransMails that will trigger a financial purchase over the next 30 days in a BlueSpace system deployed for ERP, then the user can search for TransMails that will cause a purchase over the defined date range.
- If a mail is flagged for follow up by the sender, it is more likely to be of significant relevance to the user.

5.3.3 Mail data collaborative relevance

Other scenarios leverage the relational structure of BlueSpace to determine the 'communication context' of the TransMail. These scenarios rely on both history submessages (older submessages in the chain of the TransMail), and future submessages (submessages written later as part of other TransMails that contain the submessage), as discussed in Section 4.2.

- If a TransMail is received by a user as a reply to a TransMail they themselves sent, then a submessage in the history will map to the 'current submessage' (the one at the top of the chain) in a TransMail in their sent folder. In this case, the incoming TransMail is likely to be of significant relevance.
- If a particular submessage appears many times in a user's inbox, then a new TransMail containing this submessage is likely to be of relevance to the user.
- If a new TransMail received by a user contains one or more history submessages written by the user, then it is likely this new TransMail will be of relevance to the user.
- If historical submessages in a new TransMail were found to have high relevance ratings, then it is likely that the new submessage will have significant relevance.
- If a submessage, A, does not contain a given keyword, but many submessages that were sent in replying to and forwarding submessage A do contain the keyword, then submessage A is likely to still be relevant in a search for the given keyword.

Note that these cases do imply user actions, and thus overlap with those given in Section 5.4.

5.4 Mail Actions

This section has been divided into the following subsections:

1. Mail actions inputs – the user actions recorded by the system.
2. Mail actions relevance – scenarios illustrating relevance based on the user's own actions.
3. Mail actions collaborative relevance – scenarios involving other users' actions to determine content relevance.

5.4.1 Mail actions inputs

BlueSpace audits all significant user actions inside a BlueSpace system. These actions are important as they reflect the users' opinions of the content, and the value they place on it. Example of such actions are:

- A user composes a TransMail.
- A user reads a TransMail.
- A user forwards a TransMail.
- A user replies to a TransMail.
- A user deletes a TransMail.
- A user files a TransMail in a custom folder.
- A user prints a TransMail.
- A user saves a local copy of a TransMail.

- A user copies a TransMail to their clipboard.
- A user links a TransMail (or part of a submessage) to a context.
- A user finds a TransMail as part of a search.
- A user includes a TransMail in a visualization.
- A user opens or saves an attachment.
- A user forwards an attachment.

5.4.2 Mail actions relevance

Mail actions relevance relies on the user's actions to help determine content's relevance for content in all the user's mail folders – both sent and received. Consider the following user action scenarios, in which a user's own actions can be used as an input to a relevance algorithm:

- If a user receives a TransMail from another user, and immediately forwards it to three other users with comments, then it is likely that the content had relevance to them.
- If a user receives a TransMail and replies to it, then the content of the original TransMail is likely to be of significant relevance.
- If a user receives a TransMail from another user, and immediately deletes it, it is unlikely that they perceived the content to be of significant relevance.
- If a user reads a TransMail and then moves it to a custom folder out of the inbox, then it is likely that the content had relevance to them.
- If a user has read a given TransMail many times, then it is likely to be of significant relevance to them.
- The longer the user has a TransMail on screen for, the more likely it is to be relevant to them (with caveats, such as continued user interaction with the computer).
- If a user copies text from a given TransMail to their clipboard in Windows, assuming they have the rights to do so, then this will likely be more relevant than a TransMail in which they did not do so.
- If a user links a piece of content to a context, it is more likely to be of relevance to them.
- If a user always deletes content from a specific user quickly, then it is unlikely that any content from this user is of significant relevance to them.
- If a user always reads content from another specific user before reading content from other users in any given new set of new mail received into their inbox, then it is likely that content from this specific user is of significant relevance.
- If a user typically only reads short TransMails rather than long ones, then the chances are that short ones are more relevant to that user than longer ones.
- If a user never reads messages sent to a particular group or domain (that they are a member of), then any content sent to that group or domain is likely to be of less relevance to the user.
- If a user never reads TransMails tagged with a particular Jobcode, then this content is likely to be of little relevance to them.

- If a user rarely reads TransMails designated as lowest priority (or reads them last when browsing their inbox), then lowest priority content is likely to be of little relevance to them.
- If a user sends a TransMail to a lot of users, it is may potentially be more relevant than a TransMail they sent to a single user.
- If a user sends a TransMail, and then later forwards it to other users, it is likely to be of more relevance than a TransMail that was sent once but never resent.
- If a user reads a TransMail from their sent folder, then the content is likely to be of significant relevance.
- If a user saves an attachment to their local hard disk or network system, both it and the accompanying TransMail are likely to be of significant relevance.
- If a user prints an attachment or TransMail, it is likely to be of significant relevance.
- If a user sends a TransMail to their boss, it is likely to be more relevant than a TransMail to a junior secretary.
- If a user sends a TransMail with extensive formatting, it is more likely to be important/relevant to them than if they write a few lines of plain text.
- If a user has not accessed the folder in which the content is stored for over a year, then it is likely that the content is not particularly relevant.

Note that many of these relevance examples (and those in other sections) are not black and white – while the impact of some actions is obvious, others are more difficult to determine outright, and may vary between users. This is precisely why the algorithm must learn based on each user's behaviour, and then use the past to predict the future.

5.4.3 Mail actions collaborative relevance

Mail actions collaborative relevance relies on the actions of other users to determine the relevance of content in a given user's folders – both sent and received. In the following scenarios, other users' actions can be used to help determine the relevance of content to a given user:

- If a user sends a TransMail to someone, and they read it, it is more likely to be relevant (to both the receiver and sender) than if they do not.
- If a user sends a TransMail to a group of users, and most reply to it, forward it or copy it to custom folders, it is more likely to be relevant than if they all delete it immediately.
- If most other recipients of a TransMail delete it as soon as it arrives in their inbox, then the recipient in question will probably not find the content particularly relevant.
- If most other recipients copy a given TransMail to a custom folder (or reply or forward it) once they have read it, then the recipient in question will probably find the content to be of significant relevance.
- If a given user is always copying another user on their mail content (e.g. an employee copying their boss) then the content is less likely to be of relevance to that other individual.
- If a user sends a submessage to another user, but the recipient doesn't read it, then this may have an affect on the relevance of the content.

- If the vast majority of the users in a given department read a piece of content sent globally, then the content is likely to be relevant to another user in that department.
- If the majority of other users tend to read content linked to a particular Jobcode, then content attached to that Jobcode is more likely to be relevant to the user in question.
- If the majority of other users tend to read content linked to a particular context, then content attached to that context is more likely to be relevant to the user in question.
- If most users read content from a particular author (e.g. the CEO), then content from this author is typically of greater relevance than content from a different author that no-one ever reads.
- If most users reply to content from a particular author (e.g. the CEO), then content from this author is typically of greater relevance than content from a different author that no-one ever replies to.

5.5 Mail actions importance

In addition to considering content and users' actions on it, the application can also analyse the importance of particular actors and entities when determining their relative weighting in calculating relevance. Such entities include:

- Users
- Groups
- Domains
- Folders
- Jobcodes
- Contexts
- Signatures
- Other business process information
- Any back-end/automated actors (e.g. external computer systems)

In normal business operations, every employee has an importance rating for other employees. For example, it is likely that an employee would find other individuals who report to them and those to whom they themselves report to be of greater importance than individuals working in a different department.

Importance in BlueSpace is also a relative value, respective to each user (or other entity) in the system. For example, users, groups, domains and other entities all have different relative importance values describing a particular Jobcode. Ultimately it is possible to calculate an overall importance value for any given entity in the system, which describes that entity's importance to the entire enterprise (defined either as all users or all entities).

Importance for any given contact can in fact be a pair of different values. The first value reflects the user's interest in content they received from the contact, and the second value reflects the user's interest in content they sent to the contact. Whilst it might be expected that these values would be very similar, there are scenarios in which they would not be similar. If the head of HR is extremely 'chatty', he/she may have low importance when sending content to other users, but those other users may view the head of HR as being very important when they themselves are doing the communicating.

Commented [JPM1]: I'm assuming our maths ignores this right now?

Varying weightings for relevance inputs with importance means that if a user such as a boss deletes a TransMail, then this will have a greater affect than if a

junior secretary deletes it. Thus different users' actions have different affects on relevance, due to varying importance factors.

Importance has multiple feedback affects, as if user B receives a TransMail from user A, and then user B forwards it to user C who is of high importance (e.g. the CEO), then this affects the importance of all three users.

In an organisation with, say, 100,000 employees, it is likely that any given user will communicate with new contacts each week. Importance is a relative factor for each pair of users' (or entities') interactions, and is stored with respect to each user, as described above. However, when a new contact communicates with a user for the first time, the system can still make a 'best guess' about that new contact's importance based on collaborative importance. The value then changes as the user interacts with this new contact. This first 'best guess' value is important, however, as in a system with a large population, the usability would be hampered if every new contact started 'at par'.

In certain scenarios, an organisation may decide to tie importance to roles rather than specific users. This is particularly likely in highly regimented environments, such as the military, where the role is a greater determining factor than the actual human personality involved. This is discussed in greater detail in Section 5.7.

5.6 Correlated Entities

As discussed in Section 5.5, a user will often be interacting with a new entity for the first time in large system, and thus the initial values of importance are very important. Another approach to these initial values is to recognize correlated entities, described through tree structures.

For example, one HR employee is generally likely to have similar importance to another one, especially if they are of similar position in a tree structure – e.g. they both work in employee appraisal form processing. There are clearly exceptions to this trend, but in most organizations it is likely to have a positive affect on the initial states of importance.

Note that if required, branched structures – such as knowledge taxonomies for contexts, directories for users and Jobcode hierarchies – can be used as a correlative basis for importance on an ongoing basis. Thus for example, an importance change for one user would have a cascading 'ripple effect' for similar users.

5.7 Roles

The nature of user identities can be quite complex in some organizations. If a user moves position within the company, their role will likely change, so they will have greater relevance to some users and less relevance to others. To handle this, an organization can choose to integrate their HR system with the relevance algorithm. When a person moves positions, the learning sensitivity of the algorithm with respect to that user can be increased, in order that the relevance and importance values can reach their new equilibrium more quickly. The increased sensitivity can be made to decrease according to an appropriate statistical function over time, until the sensitivity is at regular levels.

If a user moves into a role that was previously filled, the relevance identity for the previous occupant of the role can also act as an input to the individual taking on the role. This again reflects the blend of role and the user's own identity that determines their importance values, and relevance of their content. Thus for example, a person moving into a new role would hopefully still attach the same importance to content received from their spouse, but would likely move to a different importance value for their new line manager.

An additional issue revolves around scenarios in which a single person has multiple roles, so they may send email from a number of user accounts. This is a frequent occurrence in the defense sector. Another complexity is that often a single role may be shared by a number of different individuals, all of whom write content from a single account. In this case, an organization may choose to treat the role as a single entity in isolation (so there is no correlation with the users writing content from the account). Alternatively, an organization may specify a blend of the person's identity and the role being used – in which case performing an action whilst signed into one account effects the values in the other(s).

5.8 Similarity

A mail environment can have 250,000 users in a single corporation, and handle over ten million messages every day. A key problem for any algorithm attempting to recognize patterns across such a large data sets is how to reduce the problem into manageable chunks. One of the ways to do this is to reduce the number of users that the algorithm uses in collaboration, and this in turn requires sampling based on some form of similarity.

Various techniques have been used for sampling data and drawing comparisons of similarity. Comparative anatomy and the debates surrounding taxonomic structuring of biodiversity into species and genera has perhaps been one of the oldest scientific disputes. Skull size, positioning, weight, common points analysis – which is the best indicator of similarity?

The same challenge exists in determining similar users for analysis in BlueSpace TransMail Relevance (as opposed to considering the entire population for collaborative calculations). This challenge is discussed further in Sections 9 and 10.

5.9 Extensions

The relevance methods described in this white paper have focused on the users' interactions with the mail system. It should be noted, however, that a corporation can decide to extend the action observations for relevance to the local file system. A user opening a file, editing a file, etc can then be made to impact the file's score in the relevance engine of the BlueSpace system.

A second extension specific to files is that of correlating relevance values together for various versions of the same file, e.g. as part of a collaborative document creation process. This is facilitated by the BlueSpace Document Management system, which tracks various file versions through the creation process using deal room techniques.

5.10 User Interaction

In addition to the actions that users perform in the BlueSpace system, BlueSpace can take direct user input to help determine relevance. For example, a user can have a 'importance slider' for each user in the TransMail system. For example, a user might make their boss 'very important' but their junior, chatty sales clerk 'rarely important' – these weightings can then be added into the relevance algorithm.

Each piece of content (submessage) can be presented to the user along with a slider that indicates its current relevance score. The user can move this slider, and apply the change in relevance to that specific piece of content, or add that change back in to how relevance is calculated across the whole user's mailbox.

To assist in providing tangible data points for the relevance algorithm, the application can also directly query the user on the relevance of content sampled from the system, as indicated in Figure 9. Note that this sampling mechanism can also be used as a way to corroborate data in an R&D or intelligence community. In the former case, researchers can be asked to rate the usefulness of a piece of content to the project they are currently working on. In this latter case, rather than asking the user to rate the relevance of a piece of content, the application asks the user to rate their estimation of whether the content is true. Sampling can be conducted in a random or structured manner.

In addition to the end user being able to affect relevance directly for their own inbox, certain 'super-user' roles have the ability to affect relevance across the entire system. In other words, a 'Relevance Administrator' can provide a boost to communications from the HR department. The reason for this is that relevance is not a purely user-centric phenomenon – a separate question relates to relevance for the entire enterprise. Most people may ignore interaction with the HR department, but that doesn't mean that the executive board regard communications from the HR department to be irrelevant.

5.11 Summary

BlueSpace assists users in navigating the deluge of mail content they receive each day, by calculating relevance ratings for content. This mechanism also impacts searching. Users can base their searches on conventional inputs, but can also use new criteria if they wish. The application ranks search results according to a relative relevance algorithm, rather than affording each result equal weighting. In calculating each relevance value, the application also takes into account the importance of each user in the system.

6 Email Comparison

6.1 Introduction

BlueSpace TransMail has a relational mail architecture, in which each piece of content written by a user is managed as a single instance object during its life in the mail system, as described in Section 4.1. The granularity of this architecture allows the application to record all actions by users in the system, ensuring that the application has audited the entire lifecycle of the content.

While regular email does not have this relational structure, the relevance techniques described in this white paper can still be applied to email at different levels:

1. Email unit – each unit of email is considered as a whole, no matter what other message fragments (other emails) it contains.
2. Relational restructuring – each email is broken down into its component messages, and the relational structure of BlueSpace TransMail is mapped to the email store.

In addition to application to historical content, BlueSpace Mail Relevance can be applied to email based on ongoing user actions through interaction with the email infrastructure.

6.2 Email Unit

Pure email unit analysis is the most basic approach to relevance in email content. It allows relevance to be based on simple criteria such as:

- The number of recipients
- Whether the user in question received the email as a To:, Cc: or Bcc: recipient
- The importance flag of the email
- The time the email was sent

This approach provides the least accurate relevance information, as the inputs are very limited.

6.3 Relational Restructuring

Regular email does not have the relational architecture of BlueSpace – each time a user replies to an email they photocopy the history into a new text string, thus making it very difficult to audit the lifecycle of the mail content.

The BlueSpace application has the ability to analyse regular email content and convert it into a relational system, similar to BlueSpace. This was first introduced in Section 4.3, and has been discussed in detail in the BlueSpace Email Structuring white paper.[xiii]

This approach is less elegant than structuring mail at the front end of the process as in BlueSpace TransMail. When the BlueSpace application structures information once it has already been sent, the criteria for search are those such as:

- Keywords and other standard inputs
- User mail flow actions – compose, reply and forward
- Collaborative user mail flow actions – compose, reply and forward

6.4 Email Infrastructure Integration

When the BlueSpace application is integrated with the email environment at the application level (as opposed to being a purely 'downstream' process), then the following criteria are also available:

- User mail management actions – read, forward, reply, delete and file to custom folder, etc
- User attachment management actions – view, forward, save to disk, etc
- Mail content path – tying together multiple instances of the same email

By employing the relational architecture for regular email, BlueSpace allows powerful searching of regular email, with a similar relevance system to that of the BlueSpace TransMail search engine. BlueSpace also layers the access rights policy model over the search results, as described fully in the BlueSpace Integrated Governance Management[xiv] model.

Integration with email systems can be achieved in two ways, which are not mutually exclusive:

- Client-side integration, in which a BlueSpace plug-in to the email client monitors end user's actions on the mail content.
- Server-side integration, in which a BlueSpace server connects to the email server and tags content with 'flags' so that it can track what happens to the content.

These two mechanisms are illustrated pictorially in Figure 7.

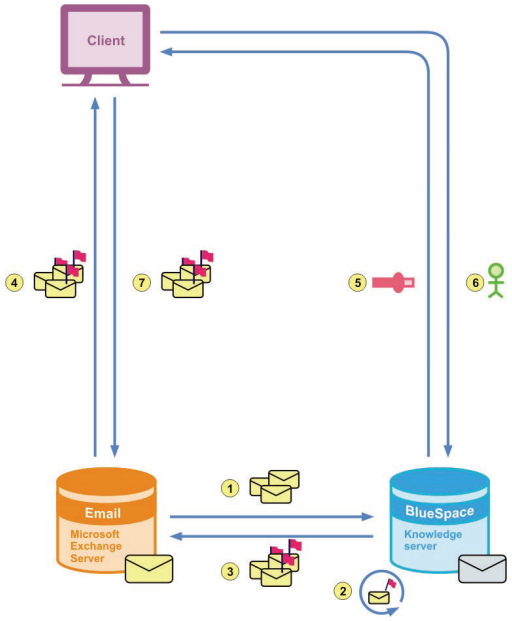

Figure 7 – BlueSpace email integration

The processes in Figure 7 are as follows:

1. New email received by the email server, e.g. Microsoft Exchange, is retrieved by the BlueSpace Knowledge Server.
2. The BlueSpace Knowledge Server adds meta-tags to each email, so that it can track that content through the lifecycle.
3. The tagged emails (represented by BlueSpace flags) are then replaced on the email server.
4. The email client, e.g. Microsoft Outlook, downloads the tagged emails.
5. The BlueSpace plug-in to the email client immediately retrieves the current relevance values for the content.
6. The user carries out actions on the content in the email client, and the BlueSpace plug-in records these actions and sends them back to the BlueSpace Knowledge Server.
7. One of the actions carried out by the end user is to send emails in replies and forwards – these include the meta-tags positioned by the BlueSpace Knowledge Server.

Note that steps 5 and 6 are a continuous feedback loop – each time the user carries out an action, the relevance value changes.

The BlueSpace plug-in to the mail client also interacts with search results, to include and reflect the relevance values in the search listings. These operations are similar to those described in Section 7.4 in the context of BlueSpace TransMail search results.

6.5 Forensics

There are many organizations involved in email forensics, and as the volume of email being sent has increased, so has the challenge for regulatory, investigative and intelligence agencies. A decade ago, a financial regulator such as the SEC might have seized tens of thousands of memorandums and letters as part of an investigation or enforcement action. Now it is routine for the SEC to subpoena millions of emails in a single action. A similar problem is faced by intelligence agencies that 'wire tap' international email flows – in which there are literally billions of emails sent each day. Internal corporate investigations and lawsuits face similar issues with data magnitude and the difficulty of selecting the most appropriate content.

There is no way that government agencies and corporations can manually read the volumes of email they now have access to, so inevitably a sample must be taken which is then passed to human investigators for analysis. Improving the automatic sampling mechanisms to select the most informative and useful content is an ongoing process.

BlueSpace Mail Relevance can be applied to regular email, with two important consequences for the forensics processes:

- **Analytics:** BlueSpace Mail Relevance supports far more advanced analytics than is currently available, by combining a more granular analysis process with the relevance algorithm. Content that meets certain criteria including relevance parameters can be flagged for human inspection, and the most important individuals in a community can be inferred automatically.
- **Similarity:** BlueSpace Mail Relevance allows determination of similar entities and users by automatic analysis of users' actions (as opposed to attempting taxonomic structuring based on automated content analysis). This is based on 'fingerprinting' different users (and entities) for similar direct and passive behaviour.

- **Dashboards:** With BlueSpace Mail Relevance, high-level representations of user communities and sub-communities can be created, based on a granular understanding of that community's interactions. It can reflect the importance of different individuals, based on the same flow analysis mechanisms for relevance discussed in this white paper. Dashboards allow analysts to rapidly gain a visual representation to a community.
- **Visualization:** The increased granular capabilities in the BlueSpace Mail Relevance system allow for advanced visualization of communication communities. An example is Force Diagram Plots of community members, based on importance, communicating frequency, etc. The same may be applied to countries, corporations, time periods, Jobcodes, keywords, as well as other categories and attributes.
- **Improved Sampling:** Combined together, these different approaches can lead to improved content sampling – both through fully automated sampling and interactive human sampling. Improved sampling is clearly a key goal of regulatory, investigative and intelligence agencies.

Note that all these techniques are essentially language neutral – an important issue for the intelligence community.

6.6 Summary

BlueSpace email search may be used as a standalone knowledge management system, for regular email systems with no TransMail penetration. It may also be used for searching legacy data once a mail system has been transferred to TransMail, or on external email content received through an SMTP gateway in a BlueSpace system with external email capabilities.

Note that to apply the BlueSpace mail search principles of relevance and user action indicators, the system does not need to restructure the content into the BlueSpace relational store. With client-side integration, BlueSpace is able to capture these actions at the application level rather than relying on inspecting mail data. BlueSpace then keeps a separate database recording user actions and links into the external content repository (mail server) through tagging mail content as users interact with it and conducting searches via a backend connector.

7 User Interface

7.1 Introduction

As first presented in Section 3.5, BlueSpace relevance and importance in mail search cause a significant improvement in the end user experience of a mail environment. BlueSpace mail search is semi-automated, in that users' actions determine the baseline behaviour of the system, but any given user can interact directly with the inputs into the relevance algorithm to better tune it to their own requirements.

This section is dedicated to describing the user interface for BlueSpace mail search, and more specifically, the BlueSpace TransMail search system when delivered through the default web interface.

7.2 BlueSpace TransMail View

There are various core pages of the BlueSpace TransMail web application that include relevance and importance aspects:

- Mail view – when viewing a single TransMail.
- Folder view – a folder listing showing many TransMails.
- Address book – a list of contacts, groups and domains.
- Preferences – a list of information

In addition, the administrative interface presents these statistics to end users.

7.2.1 Mail view

In the TransMail view, a user is looking at a single TransMail, and able to see its entire content. Various popup boxes and screen customisations are enforced on the mail view.

7.2.1.1 The reply view

An extension of this view is the reply view, as shown in Figure 8.

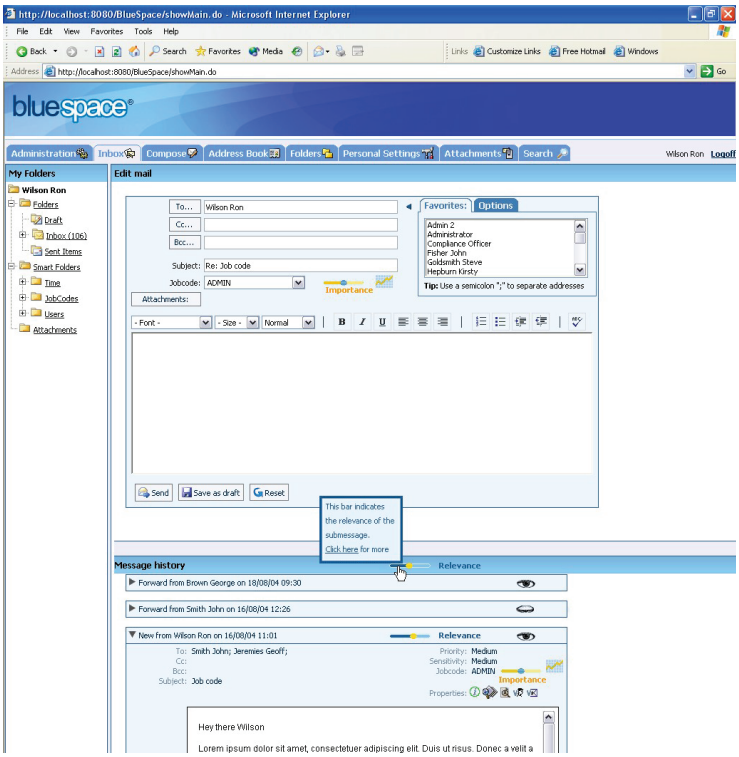

Figure 8 – BlueSpace TransMail view

There are various icons in Figure 8, and these are listed in Table 1.

Icon	ALT Text	Action
	(Popup explaining value)	Importance slider icon allows the user to change the importance value for the entity in question.
	Importance chart	Spawns the popup shown in Figure 11.
	(Popup explaining value)	Relevance slider icon allows the user to change the relevance value for the content in question.
	You have seen this submessage before	If submessage is shown collapsed, then expands it so it can be read. If submessage is shown expanded, then collapses it.
	You have not seen this submessage before	Expands the collapsed submessage.
	More information	Spawns the popup shown in Figure 10.
	Find all copies of this submessage	Shows a custom Smart Folder (dynamic) with all TransMails that contain the given submessage.

	Find submessages like this	Shows a search result page in the default preference of the user (see Section 7.4), with submessages deemed to be similar to the given submessage.
	Visualise users	Shows a popup with a visualisation of the user interaction that took place to make this TransMail, similar to Figure 2.
	Visualise mail	Shows a popup with a visualisation of the mail interaction that led to this TransMail, similar to Figure 3.

Table 1 – TransMail view icons

There are various relevance slides in Figure 8 – one for each submessage in the TransMail view, and one for the entire TransMail, taking into account the current submessage and all history submessages. Relevance sliders for history submessages are only shown when the submessage is expanded, as in the second history submessage in Figure 8.

When a user 'hovers' their mouse over a relevance bar, a popup appears, indicating why a particular relevance value is as it is – helping the user understand how the algorithm came to its conclusion. A similar popup appears for the importance sliders, discussed below.

The user can provide feedback to the algorithm, by changing the level of any given slider. When a user clicks and drags the level, similar to changing a graphic equalizer, the application shows a popup above the slider indicating the value to which the user has moved the slider. The respective values are:

- Irrelevant – 0% to 10%
- Slightly Relevant – 11% to 30%
- Relevant – 31% to 70%
- Highly Relevant – 71% to 90%
- Critical – 91% to 100%

The interface indicates to the user which submessages they have already seen before (e.g. in the case of a reply history), and which are new (e.g. in the case of a forward history). The interface shows this in Figure 8 via the state of the eye in each submessage. If the eye is open, the user has seen the submessage, if the eye is closed, the user has not seen the submessage. Thus in Figure 8, if the active user (Wilson Ron) opened the submessage from Smith John, the respective eye icon for that submessage would become open.

Figure 8 has further icons, which are described in Table 1. One icon allows the user to see a custom search folder (Smart Folder) showing every TransMail that contains the given submessage. Another searching icon allows the user to search for similar content to the given submessage, as determined by the search algorithm. Two more icons allow the user to view the lifecycle of any given submessage, spawning pop up windows that visualise its path through the system and all users that have carried out actions on it.

In addition to relevance feedback, the TransMail view can present information on importance. Jobcode importance and other custom created variables that are relatively sparse are given importance sliders as shown in Figure 8. A user can alter an importance slider, and either lock it at the new value or allow it to recalculate over time. A locked importance slider for user Smith John is shown in Figure 10.

7.2.1.2 *Mail view relevance and importance sampling*

In addition to being able to provide feedback to the algorithm by moving the relevance sliders, the user may also be requested to provide direct feedback on a sampled basis. In this scenario, the user is asked to directly rate the relevance of a given TransMail, thus presenting the algorithm with a direct relevance data point, as opposed to correlating relevance with the user's actions. This direct user questioning is shown in Figure 9.

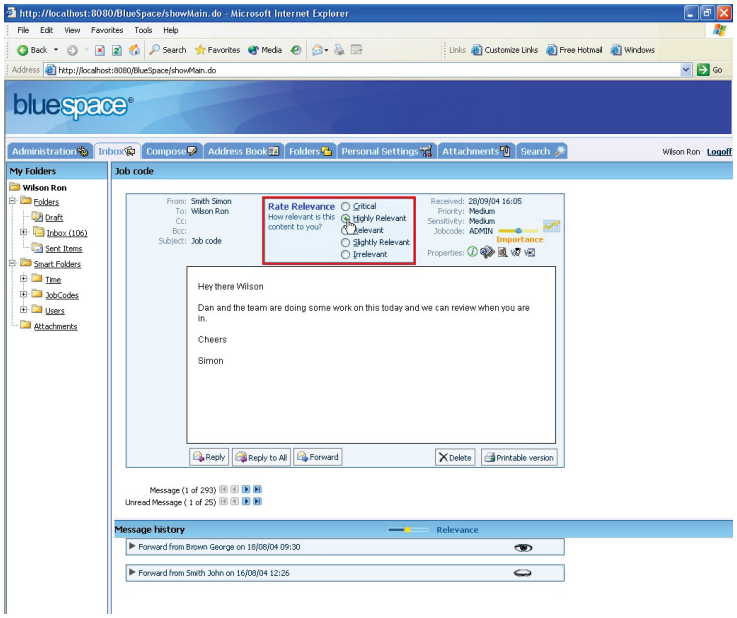

Figure 9 – Asking the user to rate content relevance

The user is not usually forced to set the relevance value before being able to leave the TransMail view, but the interface indicates that it is important for them to do so, and this would typically be reinforced by user training. It is important that a high response rate is achieved for these relevance rating questions, as the effectiveness of the algorithm correlates to the lack of bias in the survey sampling.

The number of times the user is asked to rank content can be determined based on their usage frequency (e.g. every 100 TransMail views) or absolute time (e.g. 5 times each week). A BlueSpace system administrator can determine this value, with the option to allow uses to customise it through their preferences. An administrator can also make direct user interaction mandatory, for scenarios in which this is required (e.g. intelligence communities).

The direct feedback values created when users assign relevance to content through the survey function are most useful in the early stages of a given system. They help provide absolute relevance values to stabilise the weightings given to each type of user action, as discussed in Section 7.3.4. An administrator is therefore able to either select a time-based factor to decrease the frequency of surveying users. Alternatively, an administrator can base decreases in sampling frequency on statistical results derived from, for example, the stability of the global weighting factors.

Similar mechanisms are available for direct sampling of users' ratings of importance of other users. These also function by replacing sliders, in this case importance sliders, with a survey capability.

7.2.1.3 User mail actions

When it comes to users' actions and users' importance (as well as that of groups and domains), there's a lot of data. Rather than clutter the interface with all this information by default, the application allows the user to click the details icon (the green 'i' in Figure 8) that then displays a new box that presents this information. This is illustrated in Figure 10. It shows the sender and each recipient of the TransMail, their actions on the content (forwarding the TransMail, replying etc), and the importance of the each recipient.

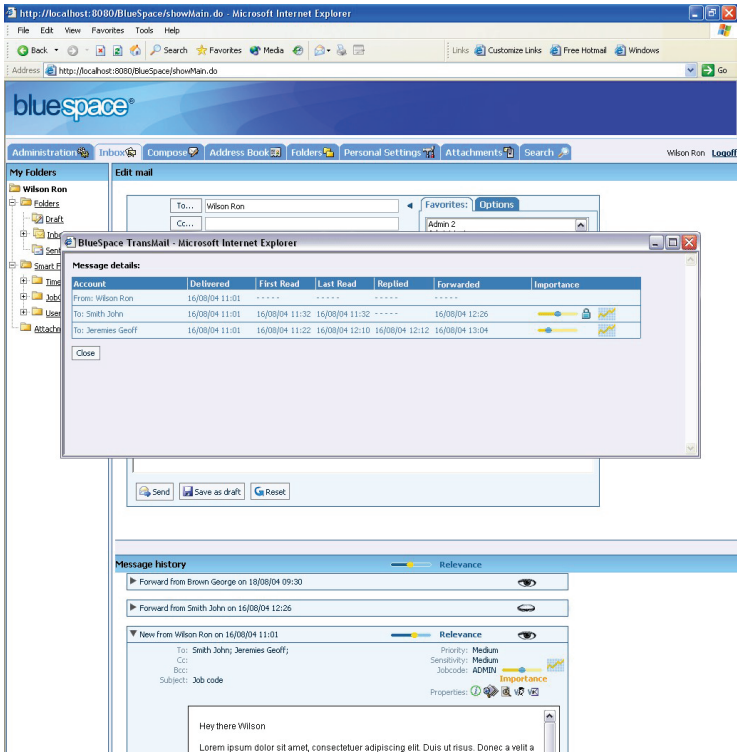

Figure 10 – BlueSpace TransMail user view

The icons in the message details popup shown in Figure 10 are described in Table 2.

Icon	ALT Text	Action
	(Popup explaining value)	Importance slider icon allows the user to change the importance value for the entity in question.
	Importance chart	Spawns the popup shown in Figure 11.
	Importance locked	Allows the user to unlock the importance value, so it can vary again.

Table 2 – Message details icons

Importance is again presented through sliders that the user can change – e.g. to tell the application that the user's boss is important. When a user makes a change, they can either lock the value with respect to the system (in order for the user to stay at that relative importance) or they can allow it to find its equilibrium value. The latter is like stressing a spring in a spring lattice – over time it will return to its lowest energy state, though this is likely to be different to the initial state as the system is continually changing over time. A locked slider is shown for Smith John in Figure 10.

7.2.1.4 Importance charts

Importance clearly changes over time – for example, when a user is interacting strongly with another user on a given project, then that other user will be of greater importance at that point than a year after the time when the project has finished. Similarly, if a user is working on a deal linked to a specific Jobcode, that Jobcode will be of far greater relevance during the deal than six months after the deal has finished. The chart icon in Figure 8 and those in Figure 10 can generate a chart showing the evolution of importance over time – for a definable period set by the user. This popup is presented in Figure 11.

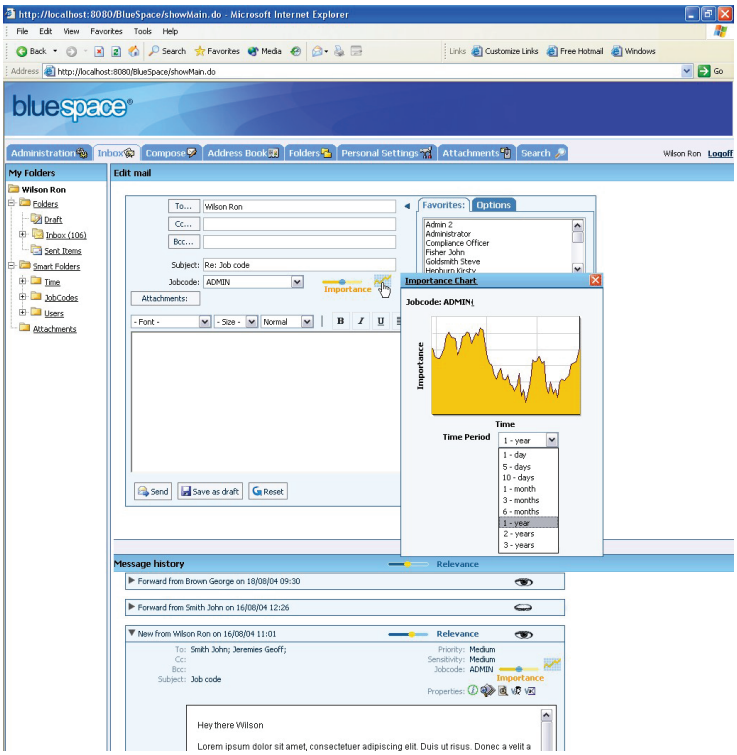

Figure 11 – Importance evolution

The evolution of importance over time is a critical concept in searching a set of TransMails, as discussed in detail in Section 7.3.

7.2.2 Folder view

The folder view is the way BlueSpace presents a TransMail folder listing to the user, showing select details of, say, 20 different TransMails at the same time. This is illustrated in Figure 12.

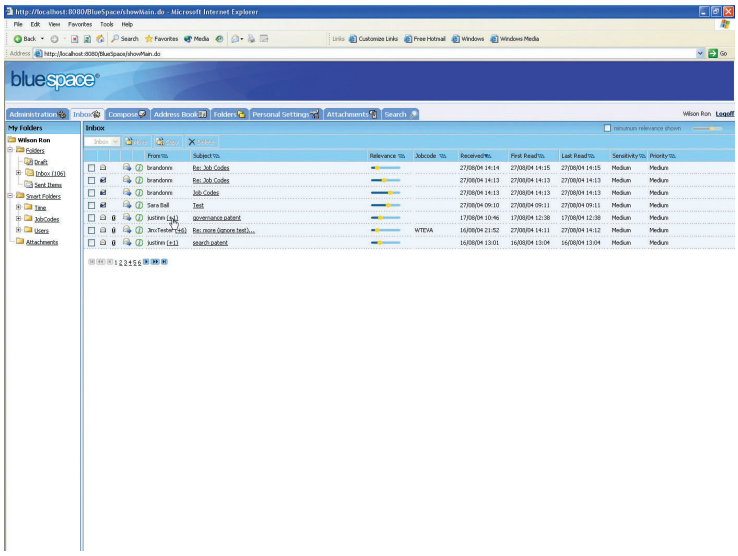

Figure 12 – BlueSpace TransMail folder view

The table in Figure 12 includes a column called relevance, which shows a relevance slider for every TransMail in the folder listing. As in the mail view in Section 7.2.1, the user can move the relevance slider to provide feedback to the application. In addition, the user can click the details icon which presents the same new box as that shown in Figure 10, indicating users' actions and importance indicators.

7.2.3 Address book

Figure 13 shows the BlueSpace address book. Each user (as well as each group and domain) is shown, and can be added to a user's favourites (those they communicate with regularly).

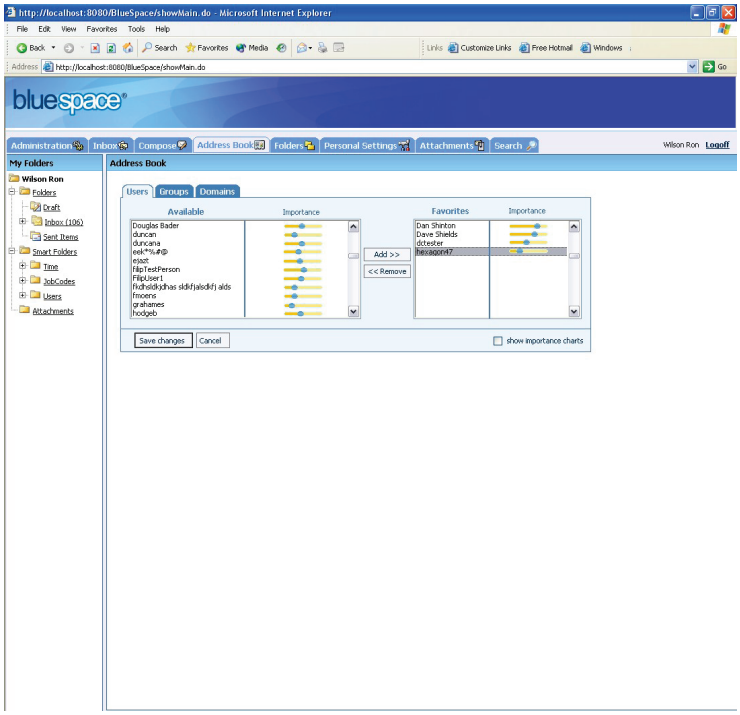

Figure 13 – BlueSpace address book

Figure 13 also shows an importance slider for each user, group or domain, and allows the user to directly interact with the importance values given to their contacts. Again, the user can either lock the value or provide it as a new start value from which the system equilibrates. Note that when user A adds user B to their favourites, the importance of user B with respect to user A is given a boost (in the algorithm).

7.2.4 Preferences

Using preferences the user can affect the different inputs into the relevance algorithm. This allows the user to tune the relevance algorithm to provide the content they are interested in. They can affect the relative strength of the different actions by users, as well as affecting the relative strength of the different entities with importance (Jobcodes, users etc). This is depicted in Figure 14.

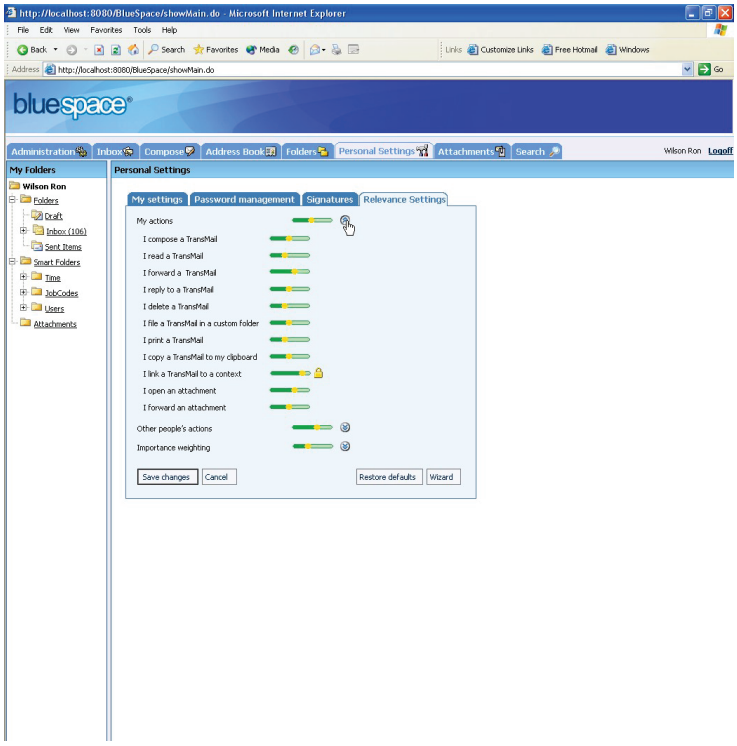

Figure 14 – Tuning the algorithm

The icons shown in Figure 14 are described in Table 3.

Icon	ALT Text	Action
	Change the significance of this action on relevance	Tuning slider icon allows the user to change the level of impact each action or entity has on the relevance algorithm.
	Significance locked	Allows the user to unlock the significance value, so it can vary again.
	Hide	Collapses a particular section of the UI.
	Show	Expands a particular section of the UI to show the respective actions or entities.

Table 3 – Algorithm tuning icons

In addition to varying the relative strengths of various inputs into the relevance algorithm, the user can also alter the importance of the different entities in the system, using an interface similar to that depicted in Figure 13 linked from Figure 14. The user can either lock the input at that value, or let it equilibrate over time – the slider for the impact of linking content to a context is locked in Figure 14.

The actions listed in Figure 14 can be customised, such as adding “I saved a local copy of a TransMail”. Such customisations can be done on implementation or by the system administrator.

Note that the mechanism of surveying users outlined in Section 7.2.1.2 provides the raw data that usually determines the weighting values. A given user's input partially determines both their own weighting values and (to a much smaller extent) the overall default weightings across the entire system. The reason for taking a content surveying approach to determining weightings is that weightings are a very esoteric factor, and it is very difficult for a typical user to determine these values directly.

In highly specialised systems, determining the weightings values can be done via a consulting exercise by the Systems Integrator (SI) implementing the system. The application has the ability to serve automated surveys to users via the weightings interface, in which users are 'walked through' various scenarios, and their responses determine the weighting values. This can be provided via the 'Wizard' button in Figure 14. The user can also restore the sliders back to their default values.

Another option for searching is to display the simple search capability, as discussed in Section 7.3, on every page of the application, allowing the user a familiar search function from wherever they are in the application. This choice appears in the "My settings" tab in Figure 14.

7.3 BlueSpace TransMail Search

BlueSpace TransMail search has been discussed extensively during this white paper, and it should be of little surprise that the search interface has far more flexibility and intelligence than any other mail search system. The search application can be divided into two areas – the search form and the search results.

7.3.1 Search form

The simple search form interface is presented in Figure 15.

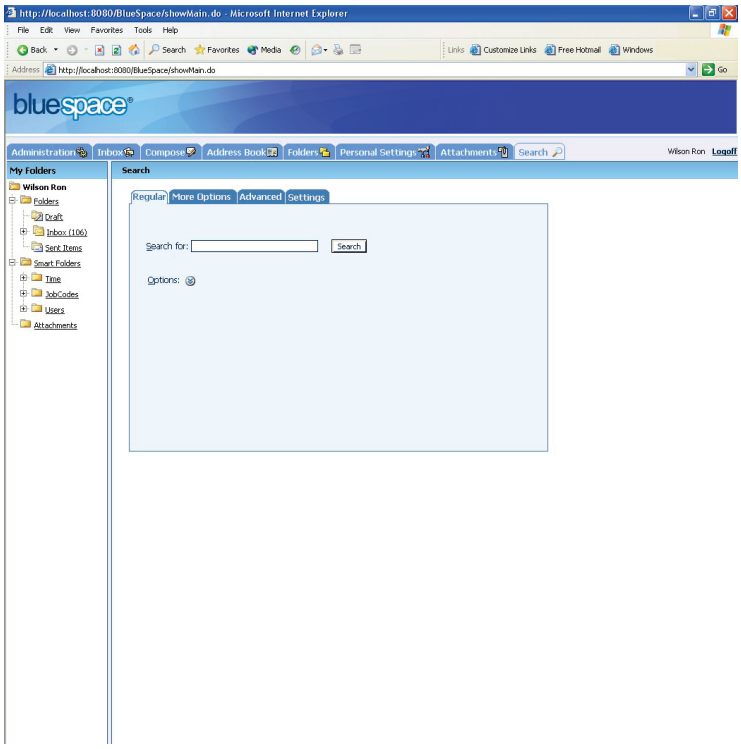

Figure 15 – Simple search interface

The tab shown is the simple search option, in which a user just enters any keywords they are interested in. The search algorithm then searches for this text in all possible criteria – users, groups, Jobcodes, message subjects, message bodies etc. This option provides an incredibly simple search algorithm for an average user.

If the user clicks the 'options' roll down in the search form interface as shown in Figure 15, the application then provides some additional search options, as shown in Figure 16. Note that the application 'remembers' when the user clicks the options roll down, and serves the search interface to them with the options already showing the next time the user goes to the search page.

The options roll down allows the user to specify which mail folders to look in, people involved, key words to search for, the date range and whether the TransMail should have attachments.

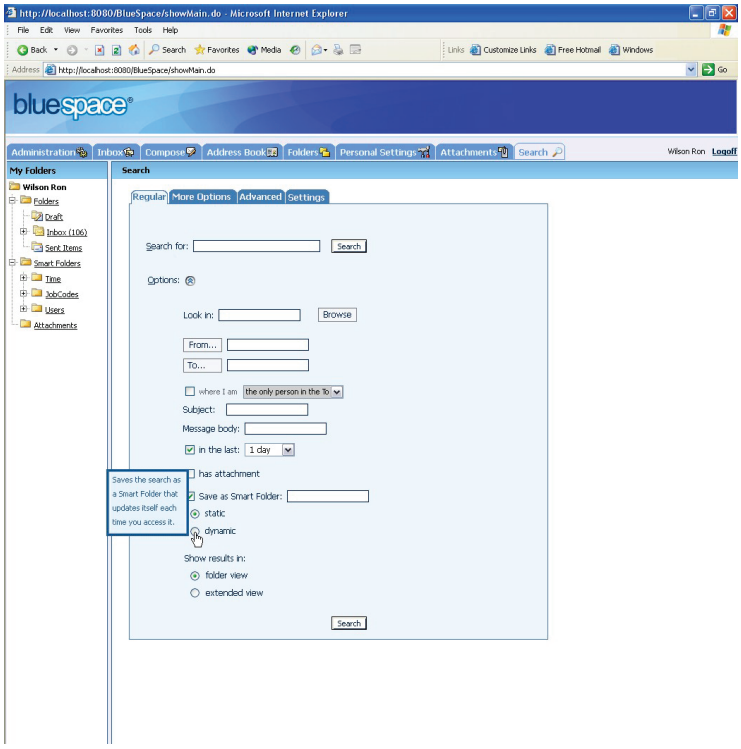

Figure 16 – Regular search interface

The search interface in Figure 16 provides the user with two pairs of options relating to what happens to search results. Each of these options has descriptive help in the form of a pop-up that appears when the user moves their mouse over the option.

A user can save any search they are doing as a Smart Folder that appears in a Custom folder in their Smart Folder tree. A static Smart Folder keeps the historical results of the search, so the user sees the same result set each time they click the Smart Folder. A dynamic Smart Folder carries out the search again each time the user clicks the Smart Folder. The user manages Smart Folders through the standard folder management interface. To edit any given Smart Folder, the user clicks the folder, and the application presents the folder's respective search query through the appropriate search interface.

The final decision for the user to make is whether they would like the folder view or extended view for their search results. These two views are discussed in Section 7.4.1.

The more options tab shown in Figure 16 provides another screen similar to that in Figure 16, but with more options for the user to choose through checkboxes, text fields, etc.

7.3.2 Advanced search

The advanced search interface is shown in Figure 17.

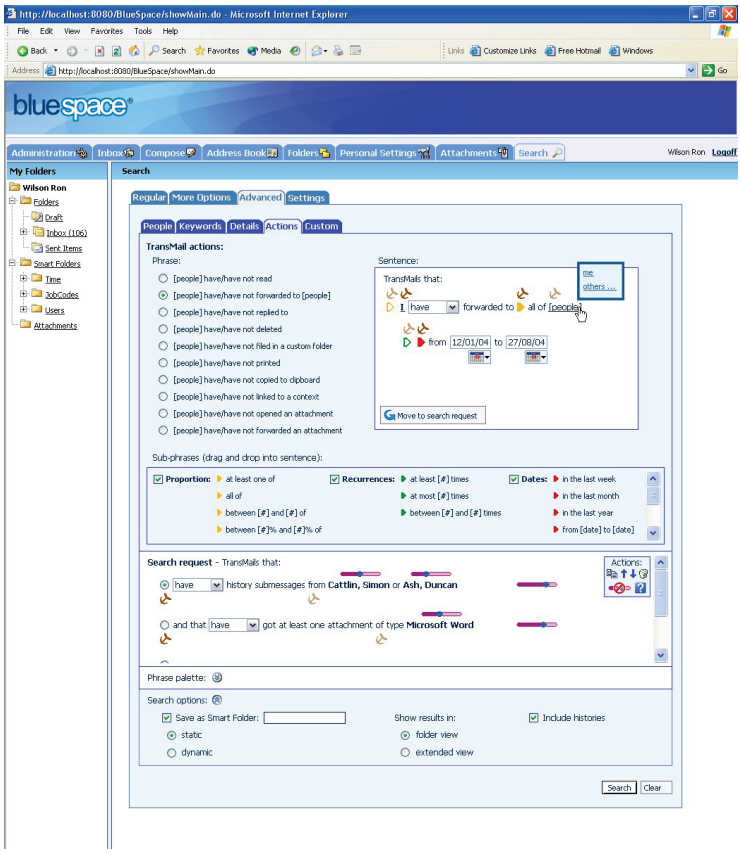

Figure 17 – Advanced search interface

Figure 17 may look rather complex, but the idea behind it is intended to be intuitive. The user can build search sentences, and then use them to do searches. The description of Figure 17 will be broken down into 5 sections:

1. Sentence types
2. TransMail actions
3. Search request box
4. Phrase palette
5. Search options

The icons shown in Figure 17 are described in Table 4.

Icon	ALT Text	Action
	Empty anchor	On click, shows a popup that indicates “You can move a phrase or sentence here by clicking a phrase anchor and dragging it here.”.
	Phrase anchor	Click and drag allows the user to move the phrase to an empty anchor.
	Empty subphrase	On click, shows a popup that indicates “You can move a subphrase here by clicking a phrase anchor and dragging it here.”.
	Subphrase	Click and drag allows the user to move the subphrase to an empty subphrase. (This icon then becomes empty.)
	Subphrase	Click and drag allows the user to move the subphrase to an empty subphrase.
	Calendar	Launches a calendar popup that allows the user to select the date.
	Change the weighting of this input to the search	Weighting slider icon allows the user to change the level of impact each action or entity has on the search request.
	Copy sentence	Creates a copy of the selected sentence.
	Move sentence up	Moves selected sentence up the ranking.
	Move sentence down	Moves selected sentence down the ranking.
	Delete sentence	Deletes selected sentence.
	Hide weighting sliders	This button hides the weighting sliders, no longer showing them in the search request box. On clicking, the icon toggles to a button without the red circle and line, that when clicked shows the weighting sliders again.
	Get help!	Launches a help and debugging tool.
	Hide	Collapses a particular section of the UI.
	Show	Expands a particular section of the UI to show the respective actions or entities.

Table 4 – Advanced search icons

7.3.2.1 Sentence types

There are five sentence types in the advanced search interface:

1. People – user, groups and domains that the mail is from and to.
2. Keywords – keywords in any field in the system, e.g. the subject or message body.

3. Details – details and properties of the mail, e.g. whether it has attachments or has been assigned a Jobcode.
4. Actions – user's actions on mail content, such as reading or replying.
5. Custom – access to a library of other sentences.

These five sentence types are presented to the user through a tabbed interface in Figure 17, allowing the user to switch between the different types as they build up their search request.

7.3.2.2 TransMail actions

TransMail actions represent the largest area of screen coverage in Figure 17, because it is that sentence type that is shown in the interface. There are three subsections within this part of the view:

1. Phrase – the area in which the user selects the type of action they wish to use in their search request.
2. Sentence – the area in which the user builds up the sentence they wish to add to their search request.
3. Sub-phrase – the area from which the user can introduce modifications to their basic phrase.

In Figure 17, the action selected is that of a person forwarding a mail to other people (be they users, groups or domains). This phrase appears in the sentence box, allowing the user to 'fill in the blanks'. In Figure 17, the user has already selected the first blank – that they themselves forwarded the mail. The user is in the process of filling in the second blank. By clicking the [people] field with their mouse, they have caused the application to spawn a pop up window. This window allows them to select the most request choice (themselves) or go to a modified view of their address book to select the people they wish.

The user can customize the phrase in the sentence box by adding in sub-phrases from the sub-phrase screen area. This part of the interface works a little like building bricks for kids. Each of the different sub-phrase types has a color coordinated arrow next to a given sub-phrase and a matching slot in the sentence box. The user can drag and drop sub-phrases into the sentence box, and then customize them as they wish. Figure 17 shows two sub-phrases have been added in – the "all of" sub-phrase and the "from [date] to [date]". The date sub-phrase requires further user interaction to specify the dates, which the user has already carried out using the associated calendar buttons. Each calendar button spawns a DHTML calendar that allows the user to specify a date. Note that date formatting is a user preference determined globally across the application via the "Personal Settings" tab.

Once a user has built a search sentence that they wish to add to their search request, they can click the "Move to search request" button, which will add the sentence to the bottom of the list in the search request box. In addition, the user can use the anchor system. Any hard anchor (shown in dark brown in Figure 17) can be dragged to a light brown (empty) anchor slot, subject to certain rules regarding how phrases can be combined. The anchor system is common to the sentence are, the search request box and the phrase palette.

Note that in order to assist the user, there are various user popup help boxes that appear when the user moves their mouse over the different parts of the interface.

7.3.2.3 Search request box

The search request box is the screen area in which the user builds up their search criteria, drawing on sentences from each of the different sentence types. Each sentence is shown, along with anchors allowing each sentence to

extended with other sub-phrases. For example, the sentence, “have history submessages from Cattlin, Simon or Ash, Duncan” can be extended at the “from” point to “have history submessages (from Cattlin, Simon or Ash, Duncan) or contain the keywords ‘sales proposition’”.

Each sentence (in both the search request box and the sentence box) can be changed between positive and negative logic. The user can change the “have” dropdown box to “have not”.

There are weightings for each sentence and each of its sentence criteria. These are shown by the purple sliders in Figure 17, and the user can move the positions of these sliders to affect the relative importance of the respective inputs in the search request. In the example shown in Figure 17, TransMails containing history submessages from Simon Cattlin would be given higher ranking than those from Duncan Ash in the search results listing. Also, the first sentence regarding history submessage authorship would be given slightly greater weighting than the requirement to have at least one attachment.

The final part of the search request box in Figure 17 is the “Actions” box. This contains icons that allow the user to manage the sentences that make up their query. The top row from left to right allows the user to copy a sentence (e.g. to search for two very similar sentences), move a sentence up the list, move a sentence down the list and delete a sentence.

The icon on the left in the second row allows the user to toggle the weighting sliders on and off. When the user turns the weighting sliders off, the ordering of the list is particularly important, as the system automatically attaches greater weighting to sentences further up the list. Note that the icon shown in Figure 17 has a red circle with a line through it indicating it is to turn the weighting sliders off. If the weighting sliders were hidden, the icon would just show the slider to indicate that it would toggle the weighting sliders back on again.

The final icon in the actions box is a “?” symbol that provides the user with help on using the advanced search interface. When clicked, the icon launches a debugging interface in a pop up that walks the user through debugging their sentence to ensure that it is consistent. This interactive debugging is highly useful in helping to ensure that the interface is usable by ordinary people, as well as being both sophisticated and powerful.

Note that if a user starts designing a search in the regular search interface shown in Figure 16, and then moves to the advanced search interface, their initial input is represented as a sentence in the advanced interface. The user can then further customise and extend it from there. This provides an easy first step for a user attempting to move from the regular search interface to the advanced one.

7.3.2.4 Phrase palette

The phrase palette was mentioned in Section 7.3.2.2. It provides a ‘dumping ground’ for the user to move phrase and sub-phrases about. It is shown rolled up in Figure 17, but the user can expand it by clicking the chevron.

The phrase palette is particularly useful to the user when transferring phrases from their custom sentence type to mix with other sentence types. The custom folder and phrase library is outlined in Section 7.3.3.

7.3.2.5 Search options

Many of the elements in the search options box are the same as those already introduced in Section 7.3.1. The user can save a search as a smart folder that appears in their Custom smart folder set. The smart folder can either be static, in which case it keeps the same result listing as when the search was first run, or dynamic, in which case the application reruns the query each time the user views the smart folder.

Search results can either be displayed in folder view or in extended view, as discussed in Section 7.4.1. An additional option for the user is whether to include the historical submessages or not. Underneath the search options box lie the “Search” and “Clear” buttons.

7.3.3 Custom phrases

The custom sentence type provides access to a library of phrases and sub-phrases. This library is typically a centralised resource held server-side across the company. The user can type in phrase fragments and search for phrases that match their desired input – bridging the gap between natural language searching and logic-based searching. The corporation can also update this library with new phrases and sub-phrases as they see fit, mapping these to new logic elements at the search level.

Examples of additional phrases include:

- “that I wrote that became part of a tree linked to Jobcode X”
- “that I wrote and were read by Clarke, John”

Other phrases act to alter the search algorithm’s action, such as:

- “that will help me find someone’s name”
- “that will help me find a historical document”
- “that will help me find out about a project”

These phrases alter the search results in such a way as to assist the user in finding the type of information they are looking for.

7.3.4 Search settings

The user can interact with the settings for the search algorithm and result ranking, as shown in Figure 18.

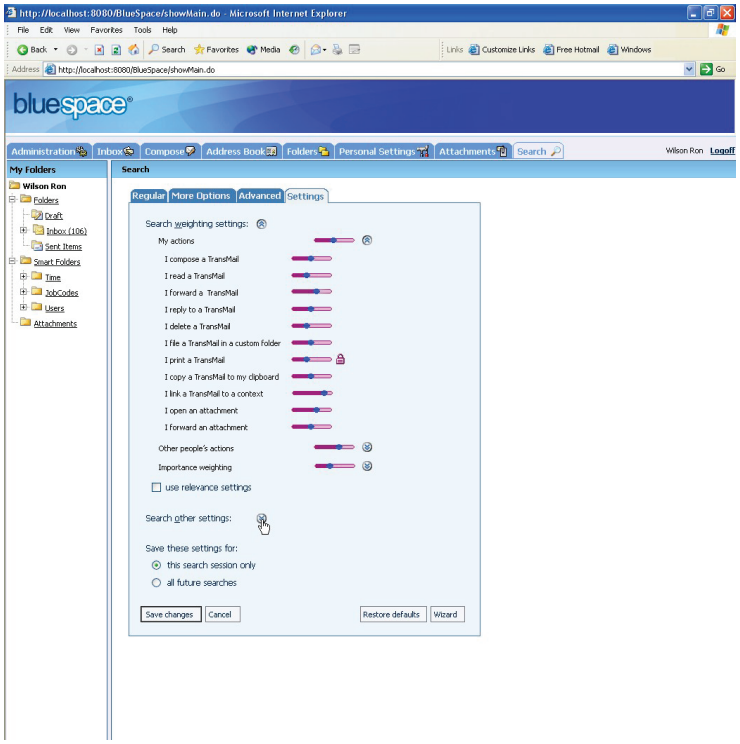

Figure 18 – Search settings

The icons shown in Figure 18 are described in Table 5.

Icon	ALT Text	Action
	Change the weighting of this action on search results	Weighting slider icon allows the user to change the level of impact each action or entity has on the search algorithm.
	Weighting locked	Allows the user to unlock the weighting value, so it can vary again.
	Hide	Collapses a particular section of the UI.
	Show	Expands a particular section of the UI to show the respective actions or entities.

Table 5 – Search settings icons

Figure 18 closely resembles Figure 14, as the user can change the relative strengths of the inputs to the search algorithm in just the same way as they can for the relevance algorithm. The user has the option of tying the weighting settings to be the same as their relevance settings.

The user has additional options they can change under “Search other settings”, including:

- The number of results per page in both search result views

- The information to include in the extended view
- The columns to include in the folder view

In addition, the user can add in new buttons that allow them to further alter the search results. These are accessed via a library similar to that described in Section 7.3.3. Examples of the functionality of these custom buttons includes:

- “Limit it to mail I read”
- “Show me mail read by more people”
- “Show me more recent content”
- “Let me change the weightings” (presents the user with a pop up to change the weightings of their search criteria)

A user can enforce any changes on either a session basis (during a particular log in session of searches) or for all future searches. This option is important, as it allows the user to increase the weighting towards specific criteria for a given search in which those criteria are important.

7.4 BlueSpace TransMail Search Results

7.4.1 Search results

Most email clients always return search results in a view resembling a mail folder. The use of this representation dates back to the origins of email, when it was designed by technologists for technologists, and reusing an existing graphical interface (the folder listing) was easier than designing another one.

The primary reason for the prevalence of the folder view for search results from mail is the wide adoption it received in email’s early years rather than its suitability for the purpose. Today there is a surprising lack of innovation in the way search results are presented to end users, just as the search mechanisms and algorithms are archaic.

BlueSpace TransMail search has two views for search results:

- **Folder view:** A representation based on the way users are accustomed to viewing search results.
- **Extended view:** A representation more closely resembling the way users are accustomed to viewing web search results.

Providing both views and then allowing the user to choose which they prefer as they conduct their search ensures user satisfaction. The application ‘remembers’ the user’s previous choice, and defaults to this view each time the user conducts a new search.

7.4.2 Folder view

The folder view of search results is shown in Figure 19.

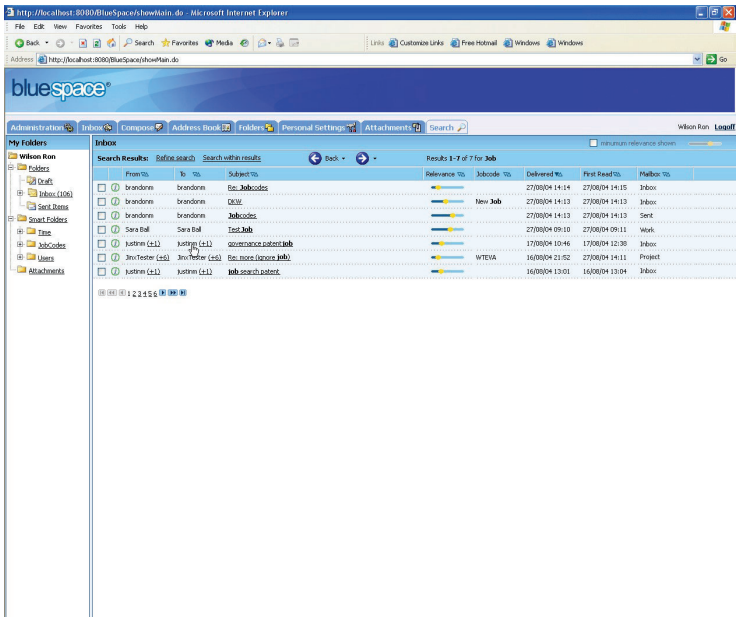

Figure 19 – Folder view search results

The icons shown in Figure 19 are described in Table 6.

Icon	ALT Text	Action
	Back to previous search	Takes the user back to the previous search they performed.
	Forward to next search	Takes the user forward to the next search they performed (only available when the user has already gone back in the search history).
	(Popup explaining value)	Relevance slider icon allows the user to change the relevance value for the content in question.
	More information	Spawns the popup shown in Figure 10.

Table 6 – Folder view icons

The view is intuitive, and designed to be such. The table indicates the author and recipients of each mail, the subject, any designated Jobcode, key dates (delivered and read) and the mailbox the TransMail resides in. In addition, the table includes an information icon that allows access to the same popup box as that shown in Figure 10, and relevance sliders for each TransMail returned by the search.

The row above the column headings includes some useful capabilities for the user. The user can refine the search and can also search within the results they have already been presented with. 'Back' and 'forward' buttons allow the user to move up and down their search history with a single click. The user can sort each column ascending or descending to further customize the results view.

7.4.3 Extended view

The extended view of search results is shown in Figure 20.

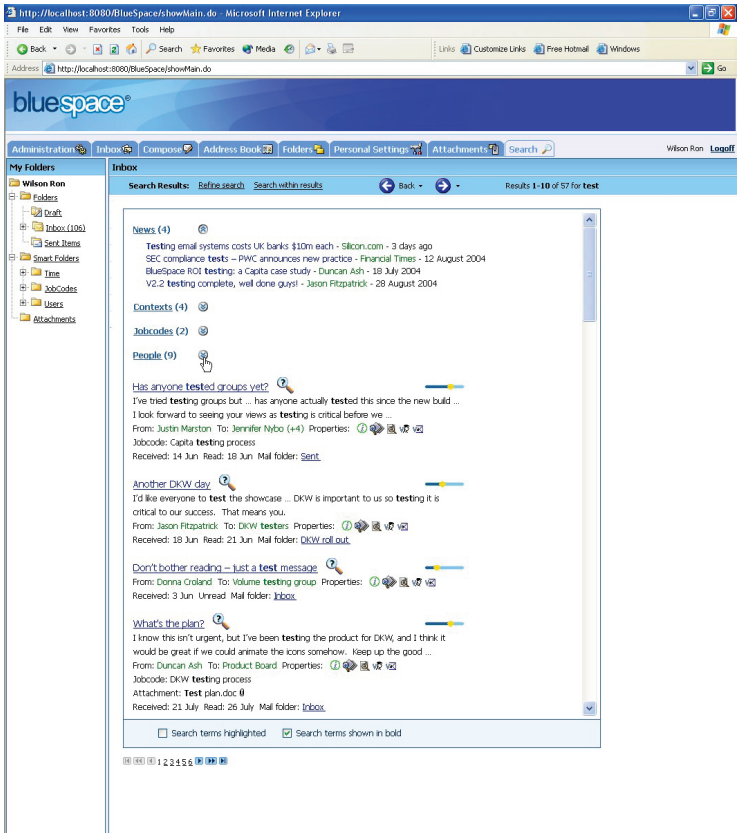

Figure 20 – Extended view search results

The icons shown in Figure 20 are described in Table 7.

Icon	ALT Text	Action
	Back to previous search	Takes the user back to the previous search they performed.
	Forward to next search	Takes the user forward to the next search they performed (only available when the user has already gone back in the search history).
	Hide	Collapses a particular section of the UI.
	Show	Expands a particular section of the UI to show the respective actions or entities.
	Why was this found?	Launches a popup indicating why a given piece of content was shown as part of the search results.
	(Popup explaining value)	Relevance slider icon allows the user to change the relevance value for the content in question.
	More information	Spawns the popup shown in Figure 10.
	Find all copies of this submessage	Shows a custom Smart Folder (dynamic) with all TransMails that contain the given submessage.
	Find submessages like this	Shows a search result page in the default preference of the user (see Section 7.4), with submessages deemed to be similar to the given submessage.
	Visualise users	Shows a popup with a visualisation of the user interaction that took place to make this TransMail, similar to Figure 2.
	Visualise mail	Shows a popup with a visualisation of the mail interaction that led to this TransMail, similar to Figure 3.

Table 7 – Extended view icons

Figure 20 bears significant resemblance to web search results, as was indicated in Section 7.4.1. As in the folder view for search results shown in Figure 19, Figure 20 includes options to refine the search, search within results, and move backwards and forwards through search history.

The “News” content set is the results of searching a corporation’s knowledge base and intranet site – not TransMails. The first two items are links to external content that the search spider found on the intranet site, whilst the second two items have been written by other employees as part of the corporate knowledge base. The search results in BlueSpace are highly customisable, and additional content feeds can be placed into these top items. The corporation can assign content feeds to each user through the administrative interface, or can allow the user themselves to choose different content feeds. This functionality can be acquired as a module that acts as an extension to the BlueSpace Integrated Governance Management module.[xv]

The next three items are shown rolled up, but the user can click the chevron to display each set of search hits. The content makes up these categories is shown below:

Contexts (4)

- Development
  - Denver
    - **Test**
  - UK
    - UI **test**
- Sales
  - TAM resources
    - ROI **test**s
    - Deployment **test**ing

Jobcodes (2)

- Capita deployment
  - Capita **test**ing process
- DKW deployment
  - DKW **test**ing process

People (9)

- Users
  - John **Test**erea
  - **Test**in Cozonack
  - Suzanne E**test**e
- Domains
  - Denver **Test**
  - UK **Test**
- Groups
  - Capita **test**ers
  - DKW **test**ers
  - UI **test**ers
  - Volume **test**ing group

Each of these sets constitutes a set of entities in BlueSpace, and the occurrences of the search term within them. The number shown in brackets denotes the number of search hits in each category.

Below these categories in Figure 20, the search results list hits within TransMail content. Each search hit highlights the content that caused the item to appear in the search results, with the following display schema:

- The subject of the submessage
- A magnifying glass containing a question mark
- The relevance of the submessage
- The body text showing the instances of the search term

- The people involved – sender and recipients
- A set of property icons identical to those in Figure 8
- The Jobcode or attachment name if they are part of the search hit
- The date the mail was received and read
- The mail folder in which the respective TransMail appears

Clicking a user (or group or domain) name causes the application to display the respective smart folder of the searcher's interactions with that population entity. Clicking a Jobcode, context or other entity has a similar effect.

Clicking the magnifying glass with a question mark inside spawns a pop up window indicating to the user how the result matches their query. This allows the user a quick way to understand why the result would be of relevance to their query, and how the application arrived at that decision. It also allows the user to provide feedback to the application (and thus algorithm), by indicating whether the result was what they were looking for.

Note that the relevance slider for the submessage with the subject, "Don't bother reading – just a test message" is quite low given that this is the first page of the search results. This highlights the fact that the relevance sliders are not respective to the search being conducted – they are absolute based on the output of the relevance algorithm. Relevance in search, described in Section 5.2, is used to determine the ranking for the search results (though may also be displayed as a user preference). This test submessage was deemed to have low relevance as few people read it, given it was a test and not real content. However, the fact that the body of the submessage contains many instances of the word "test" helped drive the submessage up ranking of the search results given the query was for the word "test".

The search result, "Don't bother reading – just a test message" does not show the instances of the search term in the message body. This is because the submessage is unread, and the governance engine in BlueSpace does not allow partial reading of a piece of content in a user's incoming TransMails. The user must click the submessage to read it if they wish to view the content.

There are two options for search term highlighting. That shown in Figure 20 display the search term in bold to help the user identify the terms. Another option (at the bottom of Figure 20) is to highlight the search terms – this is particularly useful for searches for multiple terms, as it highlights each term in a different colour.

The language of this section has indicated that each search hit is in fact a submessage, not a TransMail per se. It is up to the user to decide whether they wish to only search 'current submessages' as part of TransMails, or whether they wish to include historical submessages. This option was discussed in Section 7.3.2.5.

The page numbers at the bottom of the screen in Figure 20 are outside the iFrame containing the search results, and allow the user to move between pages of results.

7.5 Summary

This section has overviewed the search interface in BlueSpace with respect to the end user – how they see it. It contains a number of novel features that improve the user's search experience. By introducing real-time interaction with a relevance algorithm, mail content can be leveraged as the knowledge base it has become rather than just a 'content skip' in which no-one can ever find what they're looking for.

8 Blogs

8.1 Introduction

Blogs are an upcoming area for communication on the internet, and follow the 'smart pull' as opposed to push model. In essence, a user subscribes to a blog like a newsletter, and this data is then automatically pulled into their user interface by the application client (e.g. Microsoft Outlook).

There are a number of advantages for blogs over email for certain types of communication, such as:

- They are automatically structured by author or other criteria.
- They do not block up the inbox like newsletters.

In many ways, the popularity of blogs can be seen as a user backlash against mail overload.

8.2 Relevance In Blogs

While this white paper has focused on applying the BlueSpace relevance system to a mail environment, the same system can be applied to blogs. The actions of users can be recorded, the relevance of given blog listings calculated, and the importance of the various entities involved inferred.

BlueSpace relevance in blogs has been applied to both the BlueSpace blogs model and also to other open-standards blogs and blog interfaces using a plug-in approach similar to that described in Section 6.4 and illustrated in Figure 7.

For brevity, relevance in blogs is not described in full, as it is highly analogous to that in mail environments.

9 Mathematical Algorithm Framework

9.1 Introduction

The delivery of the TransMail system of allocating relevance and importance requires the continuous development of a highly complex mathematically based system. Algorithms are required that continually update a set of inter-related values, based on user action, so that everyone's actions affect everyone else. This has some complex implications:

- Each user/post has a different individual set of importance and relevance criteria.
- TransMail also requires the input of user-set values for the relevance and importance criteria, which both correct the values for the user concerned and also offer a check on the effectiveness of the particular algorithms being used.

Note that for organizations with many posts attached to user teams, such as some defense agencies, BlueSpace can allow users and posts to be treated separately with varying levels of inter-correlation.

This present section sets out the basic mathematical framework within which the detailed mathematical model is being developed. The full working of this model is not presented in this white paper for brevity, due to the complexity and extensive tuning involved. Instead, a simplified exposé is given, which illustrates the nature and working of the core relevance algorithm.

9.2 Aim Of Relevance Formula

Supposing that we have a set of people $a, b, c, \ldots k, \ldots$ involved in sending and receiving TransMails

The key aim of the relevance system is to produce two values for any recipient $\boldsymbol{k}$:

(1) The relevance ranking of an incoming TransMail x to user $\boldsymbol{k}$ before it is opened: ${}^{x}\boldsymbol{R}_{k}^{I}$ *its incoming relevance.*

(2) The relevance ranking of a TransMail x to user $\boldsymbol{k}$ after it is opened and read: ${}^{x}\boldsymbol{R}_{k}^{P}$ *its post-reading relevance.*

This is not exactly the same as incoming and stored relevance, because some TransMails may be stored without being opened. In fact three different levels can be identified: (1) Relevance on arrival before being seen (2) Relevance after being opened and read (3) Relevance after been seen but dealt with unopened. This, however, will be ignored for this algorithm overview.

The ultimate aim of the system can be expressed as follows:

> "If the recipient $\boldsymbol{k}$ were to open a set of incoming mails and assign relevance scores and rankings, these would correlate highly with both ${}^{x}\boldsymbol{R}_{k}^{I}$ and ${}^{x}\boldsymbol{R}_{k}^{P}$."

9.3 Factors In Relevance Formula

Factor sets $\boldsymbol{F}^{i}_{j}$ and $\boldsymbol{F}^{p}_{j}$ indicate factors that affect incoming and post-reading relevance. $\boldsymbol{F}^{i}_{j}$ are factors known for **i**ncoming mail, and $\boldsymbol{F}^{p}_{j}$ known as a result of the recipient action, ie they are **p**ost-action**.**

There are three lists of attributes that influence the TransMail's relevance to a receiver user, into which the F factors can be categorised.

The first are indicator functions, ${}^{x}D^{i}_{i}$ and ${}^{x}D^{p}_{j}$, which are dichotomous, ie are simply "on" or "off", depending on whether the attribute is "favourable" or not. This includes attributes such as:

The General Factors ${}^{x}D^{i}_{i}$ (available for both ${}^{x}R^{I}_{k}$ and ${}^{x}R^{P}_{k}$)

> "Flags for follow up – the sender indicates the TransMail requires action by the recipient".

Factors ${}^{x}D^{p}_{j}$, which are available only for ${}^{x}R^{P}_{k}$

> "The user replies to the TransMail" *[If so then D=1 if not then D=0]*

A second set of variables ${}^{x}C^{i}_{j}$ and ${}^{x}C^{p}_{j}$ are values on an ordinal or even a cardinal scale, eg:

General Factors ${}^{x}C^{i}_{j}$ (available for both ${}^{x}R^{I}_{k}$ and ${}^{x}R^{P}_{k}$)

> "the ratio of others who received it to the number who opened it"
>
> "the priority = the urgency tag given it"

Factors ${}^{x}C^{p}_{j}$ available only for ${}^{x}R^{P}_{k}$

> "how many times the user reads the TransMail"
>
> "to how many people the user forwards the TransMail"

These are placed into suitable functions $f({}^{x}C^{p}_{j})$ that are between 0 and 1.

Within the general factors referred to above, in an instance in which this is the first TransMail received from this source by a particular user, the concept of Mail Actions Collaborative Relevance is particularly useful. This concerns any actions done on this TransMail by others whom the recipient considers to have high local importance (see also below),

Some of these factors may also "decay" over time. It can be difficult to determine this without some indication from the user, but obviously there are some TransMails connected with projects that, once completed, leave the user unlikely to wish to reconsult the TransMails. Some feedback may be useful connected with a particular job or project code. There are occasions when the relevance of incoming TransMail is highly affected by the life cycle of a project. This means that there it can be useful to include a time related factor of importance that has a discontinuity break when the project is completed. This would mean that the relevance could be multiplied by a time-related factor ${}^{x}G_{k}(t)$. For example ${}^{x}G_{k}(t)$. can reflect relevance of a project up to a deadline time, t_0, beyond which the relevance begins to decay:

$$ {}^{x}G_{k}(t) = \begin{cases} 1 & t \le t_0 \\ e^{-\lambda(t-t_0)} & t > t_0 \end{cases} $$

where λ is a decay constant (high for rapid decay, low for slow decay).

Most stored TransMail has a time-decay-based element of relevance.

There is also a third factor in relevance, one that deals with the **importance** of the sender or forwarder. This is a complex concept in that there is both an absolute global importance but also an importance to the particular recipient. A recipient's line-manager may have low importance in the great scheme of things, but high importance to the recipient him/herself. This means that there are two factors here:

${}^{g}I_s$ is the global importance of the sender s. This number will be the same for all recipients of all TransMails from that sender at around that point in time, though obviously can change over time as the commercial scene changes. It can often depend, in particular, on the importance of the post as located in the organisation chart of an organisation.

${}^{k}I_s$ is the "local" importance of the sender s to that individual recipient k. This can depend, for example, on where within the organisation chart the sender is *relative to the receiver*, or whether the receiver is working with the sender on some particular inter-company project.

There are two elements of both global and local importance: the importance attaching to the post(s) held, and the importance that may attach to an individual *per se*. Suppose that the PA to the HR Director moves to become PA to the Marketing Director. Though she may retain her TransMail identity as an individual, much of both the global and local importances can often attach to the posts held rather than the individual.

N_x is the number of recipients of the TransMail x

Importance is again on a scale of numbers, and is put into a function eg $f({}^{k}I_s)$ which is between 0 and 1.

There are also importance factors involved in cases where the immediate sender s is forwarding mail from a previous sender v, who also has his/her own importance values. We can indicate this by ${}^{g}I_{sv}$ or ${}^{k}I_{sv}$ to indicate any earlier forwarder(s) of the TransMail.

A summary of a working formula for the relevance, of a given TransMail x is then: ${}^{x}R^{I}_{k}$

$$= \frac{\omega}{W_D}\sum_{j=1}^{n_I} w_i\, {}^{x}D_j^i + \frac{\upsilon}{W_C}\sum_{j=1}^{n_J} w_j f({}^{x}C_j^i) + \frac{\psi}{W_I}\left((w_1 f({}^{g}I_s) + w_2 f({}^{l}I_s) + w_3 \sum_{s=1}^{n_K}\left[w_1 f({}^{g}I_s / N^x) + w_2 f(gI_{lk}^{o\,m} / N x) \right]\right)$$

$$= \frac{\omega}{W_D}\sum_{i=1}^{n_I} w_i\, {}^{x}D_i^{i}i + \frac{\upsilon}{W_C}\sum_{j=1}^{n_J} w_j f(C_k^m) + \frac{\psi}{W_I}\left((w_1 f(I_g^m) + w_2 f(I_l^m) + w_3 \sum_{k=1}^{n_K}\left[w_1 f(I_g^m / N^m) + w_2 f(I_{lk}^{o\,m} / N^m) \right]\right)$$

where

n_i n_j and n_k are the number of attributes in each list

w_i and w_j and w_1 and w_2 are weightings associated with each attribute

W_D, W_C etc are the sum of the weightings in each of the three lists

$\omega + \upsilon + \psi = 1$

As it stands the formula allows the separation out of the three types of input. If preferred, the weights can be suitably readjusted and the formula can be rewritten in a simpler form.

With appropriate values for the weights, then, the relevance of a piece of mail m is a number between 0 and 1.

Local importance is generally more crucial to relevance of incoming TransMail than global importance. Even if a TransMail is received from someone very high up in importance (eg a Company Director), it is likely to have been sent to

many recipients. The relevance formula is therefore a function of the importance divided by number of recipients as indicated

9.4 Relevance Feedback Loop

As noted, a sample is collected by the system of the actual relevance scores that a user would have assigned, having looked at the TransMail, on a sliding scale that effectively works as a percentage. A "good" system would produce *a priori* relevance weightings that turned out to be near to the ones assigned post-opening by the recipient.

The formula used above is actually a linear combination of the various given values of the $\boldsymbol{D}$, $\boldsymbol{C}$ and $\boldsymbol{I}$ '**s.** It is therefore possible, given enough feedback data, to use linear regression to derive the weights which would minimise the error-squared terms in predicting the scores for relevance as given by the user. The database used is essentially a very large matrix, with a large number of zeros and can often become ill-conditioned. Specialised mathematics is needed to deal with such situations, and internal tests for absurdity (which can be thrown up when linear regression matrices are ill-conditioned) inbuilt.

This analysis enables two things. First, different recipients who feedback can have their own individual weighting estimated. Secondly, if patterns of weightings emerge for different types or levels or importances of user, then global best estimates of weightings on a user category basis can be refined over time and these can be useful particularly in assigning initial weighting to a new user.

9.5 Importance

The factors labelled $\boldsymbol{D}_j^m$ are things that either happen or don't happen eg "does the recipient reply?" and $\boldsymbol{C}_k^m$ known measurements eg "to how many users does the recipient forward it?" Their values are not readjusted in the light of the success or otherwise of the prediction of relevance values for incoming TransMails – only the *weightings* given to them are adjusted using the linear regression model as above.

The importance factors ${}^g\boldsymbol{I}_s$ and $\boldsymbol{I}_l^m$ are different, because they are not clearly ascertainable variables but are (estimated) parameters constructed within the model itself. Here the values of these factors are affected by the TransMail traffic that has occurred. For example the importance of a user is affected by the receipt of TransMails from other users – his/her importance perturbs or is shared by the importance of the other users. This interactive process must be taken into account before the importance can be fed into the equations.

There is, for user i, some true global value ${}^{\gamma}\boldsymbol{I}_s^{\boldsymbol{\Omega}}$ to which ideally the system should converge. In practice, of course this will never be static (importances are constantly changing), but it is a useful idealisation. Initially a starting value of ${}^{\gamma}\boldsymbol{I}_s^{\boldsymbol{0}}$ will be set globally, based on judgements of levels of responsibility within the organisation. The ideal is that starting from ${}^{\gamma}\boldsymbol{I}_s^{\boldsymbol{0}}$ the system will converge over time to a value near to ${}^{\gamma}\boldsymbol{I}_s^{\boldsymbol{\Omega}}$

It is also necessary to add in a factor that enables importance to be carried over from posts. If user-A moved from post-S to take over post-T, then user-A's global importance will be a function of their previous importance whilst in post-S, and the importance the previous holder of post-T had. Likewise local importance will also be a function of post as well as individual. This has been omitted from the mathematical description below for simplicity at this stage.

Importance attaches to the sender, but also to the Jobcode and context, which are also (passive) entities within the system. The Jobcode type carries an importance value alongside the sender. This added complexity means that actually the relevance formula has to contain weightings of both types of importance – sender and job code type and perhaps some other factors.

9.5.1 Factors affecting global importances over time

(1) **<u>TransMails received:</u>** Receipt of a TransMail from a more important sender increases the importance of recipient – dependent on how many others receive the same TransMail. "Shared importance" therefore depends on ${}^{r}I_i/N_i$ where N_i is the number of recipients of the i th TransMail. The effects depend both on the number of TransMails and the average of the shared importances.

The factor affecting the global importance at time k of a user is given by:

$$f\left(\sum_{i=1}^{m} {}^{r}I_i^{k-1} / N_i \,;\, \frac{1}{m}\sum_{i=1}^{m} {}^{r}I_i^{k-1} / N_i\right)$$

(summed across the m senders in time k-1 to k)

(2) **<u>Results of TransMails sent</u>**: A user whose TransMails are never opened or forwarded will decrease in importance, whereas one whose TransMails are usually read becomes more important [this may also stop people sending unnecessary TransMails!]. If those of greater importance read and deal with TransMails this can also increase the user's importance – unless it is always the same recipient (as would be the case of a PA TransMailing his/her boss regularly). Safeguards concerning the latter may be reviewed, though in some instances a PA acting for a more important person may need a higher importance rating if they regularly act on that person's behalf.

Again the effect of this factor is a function of both the number and average importance of the recipients and the results. Importance sharing of those who opened and read the TransMail is a function also of the time spent with it open, relative to the length of the TransMail itself:

Those who opened it: $f\left(\sum_{i=1}^{m} {}^{r}I_i^{k-1}(t_i / L_i);\frac{1}{m}\sum_{i=1}^{m} {}^{r}I_i^{k-1}(t_i / L_i)\right)$

where t_i is time spent and L_i a function of its length (obviously an important one-line TransMail can be read in twenty seconds) giving an "attention ratio" of t_i/L_i

For those who read it, there is also an effect depending on whether they:

(i) replied to it

(ii) deleted it without reply (this could have a negative effect)

(iii) stored it

(iv) forwarded it.

This can be indicated by an A representing action (and various other actions are included as described in section 5.3.1-2 above):

$$f\left(\sum_{i=1}^{m} {}^{r}I_i^{k-1}(A)\,;\,\frac{1}{m}\sum_{i=1}^{m} {}^{r}I_i^{k-1}(A)\right)$$

This can be countered by a negative effect on importance of recipients who either deleted (del) it or stored it without reading it (no).

$$- f\left(\sum_{i=1}^{m} {}^{no}I_i^{k-1} ; \frac{1}{m}\sum_{i=1}^{m} {}^{no}I_i^{k-1} ; \sum_{i=1}^{m} {}^{del}I_i^{k-1} ; \frac{1}{m}\sum_{i=1}^{m} {}^{del}I_i^{k-1}\right)$$

(3) **Relevance Rankings:** If recipients give received TransMails a high relevance score (when asked on the sample basis), then this will increase the importance of the sender.

If relevance reported by recipient is $R^r(T_i)$ for a TransMail from the user, then the importance is affected by a function of the average of the n such reports made during the time period k-1 to k, ie: :

$$f\left(\frac{1}{n}\sum_{i=1}^{n} f'({}^{r}I_i^{k-1}; R^r(T_i))\right)$$

This also allows feedback to the system to ensure that it converges to some kind of "realistic" estimates. Direct feedback on "importance" can be very difficult to build in because most people will be likely to have a better idea of importance *to them* than global importance. Some kind of periodic review eg from HR (at least dealing with in-house relative importances) can be built in, however, as a system develops.

A global importance formula readjusts the global importance in time k from that of k-1 using these various functions. The value α indicates how fast the global importance should change over time.

$$\begin{aligned} I_{gi}^{k} = \alpha I_{gi}^{k-1} + (1-\alpha)\Big[& w_1 f\left(\sum_{i=1}^{m} {}^{r}I_i^{k-1}/N_i ; \frac{1}{m}\sum_{i=1}^{m} {}^{r}I_i^{k-1}/N_i\right) \\ & + w_2 f\left(\sum_{i=1}^{m} {}^{r}I_i^{k-1}(t_i/L_i); \frac{1}{m}\sum_{i=1}^{m} {}^{r}I_i^{k-1}(t_i/L_i)\right) \\ & + w_3 f\left(\sum_{i=1}^{m} {}^{r}I_i^{k-1}(A) ; \frac{1}{m}\sum_{i=1}^{m} {}^{r}I_i^{k-1}(A)\right) \\ & - w_4 f\left(\sum_{i=1}^{m} {}^{no}I_i^{k-1} ; \frac{1}{m}\sum_{i=1}^{m} {}^{no}I_i^{k-1} ; \sum_{i=1}^{m} {}^{del}I_i^{k-1} ; \frac{1}{m}\sum_{i=1}^{m} {}^{del}I_i^{k-1}\right) \\ & + w_5 f\left(\frac{1}{n}\sum_{i=1}^{n} f'({}^{r}I_i^{k-1}; R^r(T_i))\right)\Big] \end{aligned}$$

These global importances are recomputed by the system every period at intervals as determined by the operators ie by BlueSpace. They are, of course, heavily inter-related and so this system of equations is solved iteratively in what is essentially a Gauss-Seidel algorithm procedure until convergence to a sensible precision. As the system is developed, there are more powerful iterative systems that can be applied.

9.5.2 Factors affecting local Importance over time

This is a comparatively more straightforward concept, at least for TransMails received from previous contacts. It is generally preferable simply to recompute this each time (or used an updated stored recomputation) rather than work by adjusting a starting value. This is based on a time-dependent function similar to that used in exponential smoothing, so that more recent actions have greater effect.

Relative importance is affected, of course, by global importance I_{gi}^{k-1}, but also by:

(1) Results of Past Received TransMails:

This is basically actions based on past receipts. These are weighted, based on a time dependent function as mentioned, so that more recent actions have greater weighting.

i. The number/percentage of times previous TransMails have been deleted or stored without opening.

$$\sum_{q=1}^{n} f_q^t(del;no)$$

here f^t is a time-based function for TransMail q, which is a function of whether the TransMail is deleted or stored unopened.

ii. The 'attention ratio' of t_i/L_i for TransMails opened, and number of times.

$$\sum_{q=1}^{n} f_q^t(t_{i1}/L_{i1};\, t_{i2}/L_{i2\cdots})$$

Other actions as listed in section 5.3.1-2 are also included.

(2) TransMails sent:

This is a function, time related, of how many times the recipient has previously sent "unsolicited" TransMails to this sender"

$$\sum_{p=1}^{n} f_p^t$$

(3) Relevance Rankings:

Any past relevance scores given by the recipient to TransMails from this sender." (a weighted average if more than one has been graded).

$$\sum_{r=1}^{n} f_r^t(R(T_r))$$

(4) Mail Actions Collaborative Relevance/Importance

When a new contact communicates with a user for the first time, the system can still make a 'best guess' about that new contact's importance, and a key element of this is collaborative importance. Suppose that user-A receives a TransMail from user-B. This is the first such TransMail, but user-A regards user-C and user-D as having high local importance, and both of these have now ranked user-B as locally important. This will increase the initial "best guess" of local importance the system ascribes to user-A for user-B. This implies a formula as follows:

$$\sum_{c=1}^{n} f(I_l^{c\ for\ sender};\, I_l^{resipient\ for\ c})$$

(5) User Importance Determination

Within the window when a TransMail is opened will be a slider showing the local importance of the sender as calculated by the system. The recipient will be able to adjust this if (s)he feels that it is inappropriately low or high. If this is the first time a TransMail has been received from this sender, then

the recipient can opt for a setting which will flash to remind the him or her that (s)he has not previously received any TransMail from this source, and that readjustment of the calculated importance of the sender might therefore be particularly useful.

The last set value from the recipient would be $I^{r\,t}_{l}$

This means that the overall figure for local importance is determined thus:

$$
\begin{aligned}
I^{m}_{l} = \alpha_t I^{r\,t}_{l} + (1-\alpha_t) \Big[& w_1 I^{k-1}_{gi} \\
& + w_2 \sum_{c=1}^{n} f\left(I_l^{c\ for\ sender};\ I_l^{resipient\ for\ c}\right) \\
& + w_3 \sum_{q=1}^{n} f^{t}_{q}(del;no) \\
& + w_4 \sum_{q=1}^{n} f^{t}_{q}(t_{i1}/L_{i1};\ t_{i2}/L_{i2\cdots}) \\
& + w_5 \sum_{p=1}^{n} f^{t}_{p} \\
& + w_6 \sum_{r=1}^{n} f^{t}_{r}(R(T_r)) \Big]
\end{aligned}
$$

If there are no values for $I^{r\,t}_{l}$ then the α_t value applies is zero.

If no previous TransMails have been received from or sent to this sender, then revised weightings will apply to the first two elements. If the sender is unknown to everyone in the “collaborative group” then a “first guess” can only be made with reference to the global importance.

In general there is less interactiveness involved in this local importance formula, and only the second term above becomes fluid. This means that, if another member in the “collaborative” and mutually locally important group acts in such a way as to increase the importance of a relatively unknown sender, then this could affect the priority given a TransMail in the in-box of a recipient. In simple terms, if user-A has a TransMail from user-B waiting unread, but that user-A’s collaborators act so that the local importance of B is increased (eg by actions done on user-B’s TransMails to them), then this could increase the assigned local importance of user-B to user-A, and so increase the relevance of the waiting TransMail from user-B in user-A’s inbox.

9.6 Content Analysis

For organizations involved in forensic analysis, investigation or signals intelligence, a greater level of content analysis can sometimes be appropriate. The relevance system has the capability to go to greater levels of granularity than submessages – resolving paragraphs of text or individual words.

This involves the system itself using spiders to identify terms within the content of any TransMail that statistically increase the likelihood that it is relevant to the recipient or person performing discovery.

Keywords of high relevance in a community of users can be automatically discovered based on statistical techniques. The frequencies of words in a set of mail content can be compared to regular (average) values, to discover the

words that are used most than average in the volume of email. The more frequent words can then be correlated to content relevance, to highlight more frequent words (compared to average values) that are used in the more relevant content. These techniques also allow for 'dashboarding' and visualization of sets of content (corresponding to communities or sub-communities), to provide insight into the nature of the community for an analyst or investigator. This is particularly useful when very large amounts of mail communications are being analysed, and content must be semi-automatically sampled for further human investigation.

9.7 Policing the System

In any given implementation, users may seek to find ways to increase their own global importance. Thus, for example, two friends on similar grades could arrange to send each other large numbers of TransMails which they would leave open for a while – each therefore increasing the other's importance. This is not a "prisoner's dilemma" game-theory situation because each can monitor the other's actions over time, and large numbers of read TransMails could increase both their importances. The advantage of this to them is that both would then get their TransMails read faster when sending to others. In another scenario, when the content analysis is advanced, senders could deliberately introduce phrases and words to get their TransMails read (although this might annoy recipients and so have a negative effect on their importance ranking in the long run).

BlueSpace is incorporating detection devices identifying this, and to penalise it where and as appropriate. In addition, this type of behaviour may be flagged to administrators inside a corporation.

9.8 The Possibility of Serious Error

It must be remembered that this is fundamentally an aid, and is never going to be a perfect model. Someone coming from outside the TransMail system, into a new post, who has never before TransMailed anyone or worked on any jobs with them, can often have an "importance" guestimated only on a basis of some input from an HR department. It is likely that such a person may in practice need to communicate in other ways to ensure that their TransMails are not simply ignored. This, of course, presently happens anyway with emails – busy people tend to simply delete or ignore emails from unknown and unrecognisable senders because most of them are probably spam anyway. If an email filtering system is used to see only emails from known senders then this will have an even more obvious effect.

9.9 Summary

This section has surveyed the mathematical model behind the relevance algorithm. Like any complex search type process, there are many layers of complexity, and it the intent of BlueSpace to continue to research and develop its relevance techniques and mechanisms. Section 10 discusses some of the implementation strategies and techniques from a programming and architecture perspective.

10 Software Implementation

10.1 Introduction

While the basic concept of BlueSpace Mail Relevance is reasonably comprehensible, the mathematics (as surveyed in Section 9) and the software implications are extremely complex.

Firstly, some definitions will be restated, while others will be created:

- **Relevance:** A value applied to content that reflects how interesting the content is to an entity such as a user (single relevance) or set of entities such as the HR department (collaborative relevance).
- **Importance:** A value applied to entities, such as the importance value one user ascribes to another user. The importance of a user can be with respect to an entity such as a user (single importance) or set of entities such as the HR department (collaborative importance).
- **Local relevance:** A single relevance value that is associated with a piece of content for a given user, and is held locally for the user's own mailbox. Note that this can be considered to be calculated from their own actions (the single component) and from the actions of other users in the system (the collaborative component).
- **Local importance:** A single importance value that is associated with one entity with respect to another, such as the importance of John to Bob, and is held locally for Bob's own mailbox. Note that this can again be considered to be calculated from their own actions (the single component) and from the actions of other users in the system (the collaborative component).
- **Global relevance:** A collaborative relevance value that indicates the relevance of a piece of content to the entire system. Note that this can be considered to be purely a function of the entities involved as the content continues, rather than attempting to include all possible future entities that might interact with the content (which would in any case not create a very useful indicator).
- **Global importance:** A collaborative importance value that indicates the importance of a given entity to the entire system. Note that this value should evolve slowly, and can be calculated as the average of the local importance values as opposed to attempting to derive every global importance value from each relevance value to reduce the computational requirements.
- **Similar relevance:** A relevance value that provides a more accurate indicator for the collaborative component of a given local relevance value than the global relevance value. A similar relevance value is tailored to the user in question by increasing the contribution of similar users to the collaborative component, rather than relying on a global value that is the same for all users.
- **Similar importance:** An importance value that provides a more accurate indicator for the collaborative component of a given local importance value than the global importance value. A similar importance value is tailored to the user in question by increasing the contribution of similar users to the collaborative component, rather than relying on a global value that is the same for all users.

Having defined these conceptual terms, this white paper will discuss the technical implementation in phases that reflect increasing complexity.

10.2 A Phased Approach

To aid explanation, the software implementation can be discussed in the following steps, mirroring the different aspects of Figure 6, adding in new capabilities with each step. Note that a corporation can also implement the solution according to these steps, with increasing infrastructure requirements (computational power, storage, etc) for each step:

1. Single relevance of content based purely on each user's actions on content in their own folders (effectively recording what each user does with each mail, and treating every mail separately). Analysis based purely on current submessage, and not retrospectively applied to historical submessages.
2. In addition to single relevance, each entity that a user deals with is given an importance value derived from that user's actions, with no collaborative aspects between users – single importance. Analysis based purely on entities in current submessage, and not changing importance values of entities in the history portion.
3. Sampling user opinions, to confirm that the relevance and importance values assigned are accurate reflections of each user's real opinions. Also allowing optional feedback via the sliders into the algorithm (as opposed to the sliders being for information presentation only).
4. Changing relevance values of historical (and implicitly 'future') submessages based on each user's actions on a current submessage.
5. Changing importance values of entities involved in historical (and future) submessages based on each user's actions on a current submessage.
6. Relevance and importance integration with search, each again on a purely single basis (not taking any collaborative aspects into the implementation).
7. Decay relevance and importance values effectively, so that older content loses its weight over time unless it is re-accessed by the user.
8. Users allowed to vary their own settings, including the impact different actions have on relevance and importance.
9. Global relevance values calculated for each piece of content, based on the overall result of each user's local relevance values. Initial relevance for content can then be taken from the global values, and global relevance can still form a component of each user's local relevance even after they have read the content. This is the first step in collaborative relevance.
10. Global importance values, again calculated for each entity based on an overall combination of each user's local importance values (effectively an 'average importance'). This is the first step in collaborative importance.
11. Collaborative search, allowing users to search other users' content, according to the rules laid down by the governance policy (see the BlueSpace Integrated Governance Management white paper[xvi] for full details).
12. Integration with HR to assist in assigning similar users, to allow an indication of when a user changes roles.
13. Correlation of groups, domains and other aliases with users, so that the system can imply a degree of change on users' other identities when an action causes one of their identities to change.

14. Integration with enterprise naming systems, such as Jobcodes, so that similar types and categories can be used in calculating importance values for related entities.
15. Calculation of each user's relevance values for content based on their own actions, and actions of 'similar users'.
16. Calculation of each user's importance values for other entities based on their own actions, and again actions of 'similar users'. An importance value is only assigned and varied once a user has interacted with an entity – the initial value is taken from the global importance value.
17. Initial importance values based on actions of 'similar users' rather than the global importance values.
18. Allow collaborative weighting adjustment in a feedback loop, inside a single organization.
19. Analyse smaller fragments, such as paragraphs and words, as subsets of the overall mail, to determine importance for these entities too.
20. Allow federation of action weighting and other system constants based on other corporations' data (duly anonymized) to improve the accuracy of the algorithm.

Various aspects involved in these implementation steps shall now be discussed in greater detail.

10.3 Computational Impact

One of the greatest challenges of the BlueSpace Mail Relevance model is the way in which a single action can have an impact on so many different values within the system. The best way to illustrate this issue is by example.

10.3.1 Scenario

Consider a TransMail sent by John to Sue, Bob and Harry. Each of the following user actions has an impact on many values in the system:

1. John composes a TransMail linked to Jobcode HR59231 to Sue, Bob and Harry.
2. Sue reads the TransMail.
3. Sue forwards the TransMail to Clarence (which in turn initiates more actions, not considered here).
4. Bob reads the TransMail.
5. Bob prints the TransMail.
6. Bob replies to the TransMail.
7. John reads the reply to the TransMail.
8. Sue reads the reply to the TransMail.
9. Harry returns from holiday and reads the original TransMail.
10. Harry reads the reply to the TransMail.
11. Harry deletes the reply to the TransMail.

10.3.2 Impact on relevance and importance values

Now consider the impact of action 5 in the above list. Bob printing the TransMail will impact:

- Bob's local relevance value for the TransMail, which will increase because he printed it.
- Bob's local importance value for the Jobcode HR59231.
- Bob's local importance value for John, as well as that for Sue and Harry.

These variations in local values then cascade into many further changes to other values:

- Bob's changed local relevance value for the TransMail will have an impact on the global relevance value and any similar relevance values for the TransMail.
- Bob's changed local importance values for Jobcode HR59231 and those for John, Sue and Harry will have an impact on the global importance values and any similar importance values for those entities.
- The change in Bob's local importance values for Jobcode HR59231 and those for John, Sue and Harry will have an impact on all the local relevance values for content Bob has sent and received that involve those entities.
- The change in global relevance value for the content will have an impact on the local relevance value of the TransMail for Bob.
- The change in global importance values for Jobcode HR59231 and those for John, Sue and Harry will have an impact on the local importance values of each of those entities for Bob.
- The change in global and similar relevance values for the TransMail will impact the local values for John, Sue and Harry.
- The change in global and similar importance values for Jobcode HR59231 and those for John, Sue and Harry will have an impact on each user's local values, as well as all the other local values (held by other users) with respect to those entities.
- The change in local importance values will impact the local relevance values of many users for large amounts of content.

This can quickly get out of control, as the cascading effect of changing one value can spiral into millions of updates to values for every action performed by the user in the system.

10.3.3 Analogy to other systems

To aid in gaining a potential approach to this problem, BlueSpace compared the challenge to two other highly computational systems:

1. **DNA folding:** DNA molecules are highly complex made up of many thousands of atoms. Often a researcher may know the 'street map' of the DNA (which atoms are directly bonded to which other atoms), from sequencing, and so can draw out the double helix picture seen in many TV documentaries. But every atom in the molecule has some degree of interaction with every other atom (due to electric charges etc). The shape of the DNA molecule is determined by the way the molecule folds (similar to protein folding). Attempting to do computational analysis to determine the shape of a given molecule is a very complex problem, and is generally approached as an iterative process to determine the lowest energy state for the molecule. These mechanisms are susceptible to incorrect answers, however, due to false minima.

2. **Solar systems:** Solar systems again involve very large numbers of entities, each with weight, volume, gravitational forces etc. Every entity has some degree of interaction with every other entity. An additional complexity in predicting the evolution of a solar system over time is the fact that it is continually changing rather than just having a single lowest energy state. The accuracy of future predictions (just like those of the weather) decrease significantly with time.

Both of these analogous challenges feature complex interactions involving many different entities, and require calculation of the lowest energy state. BlueSpace Mail Relevance has similar requirements.

10.3.4 BlueSpace relevance

In attempting to build a model for BlueSpace Mail Relevance, BlueSpace made the following observations:

- Some interactions are going to be greater than others. For example, Bob printing the TransMail in the scenario given above will have a much greater impact on the global relevance value for that piece of content than on a global relevance value for another piece of content for which Harry is one of many recipients. This has implications for the extent to which 'multi-order' (distant) impacts are fully calculated.
- The size of the data sets involved will vary. For example, Bob printing the TransMail will have a greater impact on his local relevance value for that content than on his local importance value for John, because Bob has exchanged many TransMails over a long period of time with John. This has implications for the impact of any given action.
- The desired rate of change will also vary. For example, the rate of change of Bob's local relevance value for the TransMail should be much faster than the rate of change of Bob's local importance value for John. This is also true of global values – relevance for a given piece of content should change much faster than importance of a given entity. If relevance values are not kept up to date, necessary changes from the initial values, based on users' actions on the specific content concerned, will not have been completed until after all the users have already read and dealt with the content. This has implications for the prioritization of effecting changes in relevance and importance values.
- For the same type of value, the size of the data set will affect the impact of any given action. For example, for a TransMail with one hundred recipients, the actions of the first few users to deal with the content should have a greater impact on the global relevance value for that TransMail than those of the last few users to deal with it. This again has implications for prioritization of effecting changes in relevance and importance values.
- For the same type of value, the required frequency of update decreases over time. In order to be useful to users when they are browsing content, the local relevance values of recent content must be updated frequently. Most users typically read most mail content with a few hours (or even minutes) of receiving it. If other users' actions have no immediate impact on local relevance values of the content, then the collaborative component has far less value in determining local relevance. Local relevance for older mail content, however, would not be expected to change rapidly, and thus could be updated less frequently. This has implications for the frequency of update of values (particularly relevance), because the required frequency decreases with time.

In summary then, just as distant planets will have a far lesser impact on the trajectory of a comet than planets nearby, so 'distant' users (in other departments with little interaction) will have a far lesser impact than related users who behave in a similar manner and interact regularly. This allows the time between updating some values to be far longer than the time for others, and can allow some distant interactions to be ignored altogether to reduce computational complexity.

10.4 Local And Global Values

One of the greatest challenges for an algorithm of this type is the size of matrices involved. Take the global values for relevance for a given piece of content. One approach could be to update this number in real-time or near real-time using queuing. But when a mail is sent to 5000 users, the application would need to do or queue 5000 alterations to the global relevance value for that piece of content. Taken to its logical conclusion, every significant action would require a change to a number of interconnected global relevance values. This would create an unmanageable computational load, and would also likely lead to database locks, in which the application attempts to update two related values at the same time, and then halts.

Local and global values were in part created for this reason, and the way the application updates them removes the locking problem. Local values for relevance of content and importance of entities are held at the individual entity level. Thus, for example, a user will have relevance values for all the content in their inbox, and importance values for each other user with whom they have sent or received content. These local values are updated in real-time, similar to the process BlueSpace uses to update the recipient timestamp fields.

The global values for relevance of any given piece of content and importance of any given entity can be calculated from all local values. Note that in this calculation, the local relevance values' contribution weighting to the global value can be equal or correlated to the global importance value for each entity concerned. Some of the global values can thus be updated in 'quiet-time' at periodic intervals, based on inspecting all (or a sample of) local values. Frequency of updating global values can be determined based on criteria such as the age of the content (relevance of recently authored content will change more rapidly) or entity (e.g. importance of new Jobcodes will change more rapidly).

Note that in reality, local values can be applied to both entities (single) and sets of entities (collaborative), as shown diagrammatically in Figure 6. Groups, domains, roles, departments, Jobcode types and other sets of entities can also have local values ascribed to them. Calculation of the global values can include these collaborative local values, or by use of similar values, as described in Section 10.7.

Local relevance and importance values have two components – a single component (based on that user's actions) and a collaborative component (based on all users' actions). This distinction is leveraged in the implementation model given in Section 10.6.

In communities that have very different sending and receiving behaviours, relevance and importance values can be split still further, into incidents when sending and incidents when receiving.

10.5 Calculation Frequencies

Each of the different relevance and importance values has different characteristics which help to determine how frequently the values must be updated and the mechanisms for doing so:

- **Browse relevance:** This is the main measure of relevance as discussed throughout this section. Browse relevance is based on a single and collaborative component, both of which must have readily available values when the browse relevance is required for a view in the application (e.g. a user views their inbox).
- **Search relevance:** These values are used to rank search results, and since the criteria are entered by the user at the time of performing the search, the values must be determined at run-time. For end user searching, search relevance is typically a function purely of their own browse relevance values. For more extensive data mining and collaborative searching, search relevance may be based on a set of users' relevance values. It is typically deemed acceptable by compliance users that this type of discovery request may take a reasonable period of time to perform.
- **Local relevance:** These values are the result of the single and collaborative components for a given piece of content for a particular user. They must be kept up to date to be of use to the user, being recalculated either as part of every transaction caused by each action that the user carries out, or otherwise held in a separate queuing system for rapid resolution. Note that local relevance values for recent content should be kept up to date as both single and collaborative components change. Values for older, historical content should not vary so dramatically, and thus do not require so frequent recalculation.
- **Local importance:** These values can be derived from the local relevance values. A key attribute of local importance is that once a value has been assigned based on a reasonable data set, the local importance value should change slowly. New entities interacting with a given user should have the ability for their importance value to change more quickly, as the data set is smaller. This is property of increasing sample size on average in any case.
- **Global relevance:** If global relevance values are the basis for the collaborative component of local relevance, then they must be kept up to date. If not, the value of the collaborative component in local relevance will be greatly reduced, as for example the impact of five other recipients deleting the TransMail before I read it will not affect my local relevance value until after I have dealt with the TransMail. As with local relevance, the rate of change in a global relevance value for a given piece of content should be greater the newer the content is (as the first users interact with it).
- **Global importance:** Global importance values should change very slowly indeed. They represent the overall importance of entities in the system. Note these values will again change more rapidly for newer entities than for existing ones that already have the BlueSpace equivalent of a 'credit history'.
- **Similar relevance:** Similar relevance values should be similar to global relevance values, in that they should change rapidly when content is new, reflecting the first real user validation of the relevance predictors for the content.
- **Similar importance:** Similar importance should typically change faster than global importance, but not as fast as local importance. Similar importance is based on purposefully biased samples, but still involves the views of a significant number of individuals rather than just one user.

The required frequencies for updating of these numbers are reflected in Section 10.6.

10.6 Implementation Detail

There are a number of processes involved in calculating and maintaining the relevance and importance values associated with content and entities. This section will outline the various processes that have been implemented to successfully manage BlueSpace Mail Relevance:

- Displaying the calculated BlueSpace Mail Relevance value when browsing the BlueSpace system
- Initial calculation of the BlueSpace Mail Relevance value for each piece of content
- Updating each persons local relevance value on auditable actions
- Updating the local importance values for entities involved in an auditable action
- Updating the global relevance values for each piece of content
- Updating the global importance values for each entity
- Updating the importance values of entities to specific roles
- Updating the local relevance values as a result of changes to local, role or global importance value
- Updating local relevance through manual importance changes

10.6.1 BlueSpace Mail Browsing

Users interacting with the system will be shown relevance values calculated for each message. The value shown to the user is an "effective relevance" based on a combination of the local and global relevance values.

Effective relevance = $w_1{}^{x}R_k + w_2{}^{x}R_g$

Commented [JM2]: Jon I need to re-discuss this with you again. Just been thinking about it more. Partly on terminology, and partly on model.

When considering the BlueSpace data model, the local relevance is associated with a user's mailbox (i.e. pointer to the content), while the global relevance is associated with the actual content. This means that display of the information has minimal impact on the performance of the system.

In the folder views, the user can sort by relevance in the same way they can sort by received date, subject, sender etc.

By combining the global relevance value with the local relevance value, collaborative relevance is obtained. For example, when somebody has been on vacation for a couple of weeks the global relevance value will have changed as a result of how other individuals receiving the same messages have interacted with that message. If each person has read and then deleted that message, the global relevance value will be very low allowing the individual who has been away to concentrate on more relevant messages first.

10.6.2 Initial Calculation

As described in section X: Mathematical Algorithm Framework, the incoming relevance value can be calculated using the following formula:

$$
\begin{aligned}
{}^{x}R^{I}_{k} &= \frac{\omega}{W_D}\sum_{j=1}^{n_I} w_i\, {}^{x}D^{i}_{j} \\
&+ \frac{\upsilon}{W_C}\sum_{j=1}^{n_J} w_j f({}^{x}C^{i}_{j}) \\
&+ \frac{\psi}{W_I}\left((w_1 f({}^{g}I_s) + w_2 f({}^{l}I_s) + w_3 \sum_{sv=1}^{n_x}\left[w_1 f({}^{g}I_{sv}/n_x) + w_2 f({}^{l}I_{sv}/n_x) \right] \right)
\end{aligned}
$$

There are three parts to this calculation:

- Calculate the weighted sum of Dichotomous Indicators
- Calculate the weighted sum of the Cardinal Factors
- Calculate the weighted sum of Importance for each entity involved in the message (and message history)

Each of these values is stored separately to facilitate ease of update as a user interacts with the message and impacts the relevance.

The local relevance for each user is calculated when the message is sent/received within the BlueSpace TransAct Server:

```
Calculate Dichotomous Indicator Weight

Calculate Cardinal Factor Weight

For each recipient (including sender)

   Calculate Importance Weight
```

The initial global relevance value is calculated from the mean of each recipient's local relevance value.

For these initial calculations, the dichotomous indicators and cardinal factors can only use the message details since the recipient has not yet performed any auditable actions. As such, these two calculations are very straightforward and only need to be calculated once for each recipient.

The importance weight calculation has multiple levels of complexity that can be configured based on the infrastructure and performance requirements:

- In the simplest form, the importance weight is calculated on the global and local importance of the entities associated with only the message received.
- A more complex version takes into account the importance of entities in the message history.
- A further option allows both of these calculations to include the importance with respect to the user's role.

Unlike the dichotomous indicators and cardinal factors that are based only on the information associated with the new message, the importance calculation has to look up the importance values from the database. As with other areas of the application, data caching is used to improve the performance of these lookups.

10.6.3 Auditable Actions - Local Relevance

A user's interaction with the BlueSpace system is subject to rigorous audit – the introduction of relevance makes use of the audit mechanism to ensure that the relevance values associated with a message are updated as a result of the user's interaction with that message, whether that is reading a message, forwarding the message to one or more recipients, replying to a message,

moving the message to another folder, deleting the message and even restoring a message.

Each action results in a change to a dichotomous indicator or cardinal factor and the user's local relevance value is recalculated and updated in the database.

The action may also result in a change to the local importance values of entities to the user. However, the local importance value of an entity affects the local relevance of many messages and as a result is not handled by this process (see below).

10.6.4 Auditable Actions – Local Importance

As discussed in section X: Mathematical Algorithm Framework, each action will have an effect on the importance values between a user and an entity. A user's interaction with the system not only affects the relevance associated with individual messages, it also has an affect on the importance values associated with the entities in the system.

This process deals with changes to the local importance values associated between a user and an entity resulting from the auditable actions from a user interacting with content in the system.

The impact of changes to the importance values are far reaching and requires many relevance values to be recalculated. Fortunately, it would be surprising for the local importance values to fluctuate significantly and would expect them to change slowly over a period of time. As such, the mechanism used to recalculate the local relevance scores as a result of changes in importance is handled separately in an asynchronous background process.

10.6.5 Global Relevance

Each user's interaction with the system impacts the relevance placed on the messages they interact with. The global relevance value is used to provide a collaborative element to each user's effective relevance value.

Rather than updating the global relevance value each time a user interacts with the system, which could cause database locking problems, the global relevance value is updated in an asynchronous background process that cycles through all the messages in the system.

```
For each TransMail
    Global Relevance = mean(each user's local relevance)
```

This background process is tuned to spend more time recalculating new and highly relevant messages.

Each calculation is required to access the database to retrieve the local relevance values for each user. The only database locking required is to update a global relevance value. This update is very fast and has no other dependencies ensuring that it does not cause locking issues in the main application.

Commented [JM3]: While I agree with this in principle, the frequency of update will need to be very high for recent content, otherwise the collaborative component of local relevance (definitions in 10.1) will not reflect user's actions until after all users have dealt with the content. Perhaps there is a process for recalculating the global relevance value that does it every 2 minutes for the first 10, then every 5, then every 15, then every hour. I'm not sure the queuing mechanism isn't bad for the first few reads, or in scenarios with less than say 10 recipients of a mail rather than 10,000. Or perhaps it does it in real-time for the first 5 recipients who access it, then switches to averaging. Just ideas.

10.6.6 Global Importance

This process is very similar to the global relevance process but updates the global importance value for each entity. As with global relevance the calculation is straightforward being the mean value of each user's local importance for an entity.

10.6.7 Role Importance

Role importance is an extension to the global importance process that is used to calculate the importance of an entity to each user sharing the same role.

Role importance is an optional component that allows the relevance values to take account of the importance of content to a specific role or post. If a user is moved to another role, the relevance of existing content may change dramatically. Role importance allows the relevance values to be weighted towards the role.

10.6.8 Local Relevance

Changes to local and global importance need to be propagated through to the local relevance values associated with specific content.

It would require extensive processing power to propagate every change to importance through to the relevance of all associated content. As such, this background process cycles through all content to recalculate the local relevance values for each user. The recalculation also takes into account the affect of time on the relevance associated with content allowing the relevance to decay over time.

This is an extensive process and could take a significant amount of time (and processing power) to revisit all messages. As with the Global Relevance process, this process is tuned to spend more time processing new and highly relevant messages.

10.6.9 Manual Importance Changes

User feedback on the importance of individuals is likely to have a more direct impact on relevance than a user's "typical" interaction with the system.

To handle these manual changes, this asynchronous process is configured to listen for "manual importance change" events. When such an event is received, each relevance value associated with the changed entity is recalculated.

10.6.10 Deployment Options

The background processes discussed in this section are all processor and time intensive. However, there are a number of deployment options that ensure the ongoing operation of the BlueSpace TransAct Server are not impacted by these processes.

In particular, these background processes can be deployed on additional servers or configured to only run at specific times of day in non-peak periods (for example, overnight) ensuring that end-user performance is not affected by these processes.

Where the user population is high and 24 hour operation required, additional hardware is definitely the best option.

10.7 Similarity

As first introduced in Section 5.8, similarity is an important principle for reducing the required computation power for large BlueSpace TransMail Relevance and BlueSpace Mail Relevance systems, while increasing the algorithm's accuracy.

A single global value for relevance of a given piece of content may not be the most appropriate measure of its relevance to a given community. For example, a TransMail regarding HR policies might be most relevant to the HR department, and of less relevance to other users.

To improve the accuracy of the algorithm, it is usually better to compare a user to a set of similar users rather than to the global population. There are a number of ways to determine similarity:

- Role or group based.
- Collaborative filtering
- Fingerprinting
- Cluster models
- Search-based methods
- Frequency analysis

Each of these different methods will now be discussed in turn.

10.7.1 Role or group based

The challenge of similarity can be dealt with in part by the role-based component in the effective local relevance value for each end user. Components can also be included for groups, domains and other sets of users (or other entities).

Users in the same groups or roles can be assumed to be similar. For example, if a number of users in a role receive a TransMail, and other users in that role have already viewed it and deleted it, then it is likely to be of low relevance to another user in that role.

This is not a particularly accurate mechanism for two reasons. Firstly, most organizations evolve quickly, with non-formalised, cross-departmental teams – often carbon copying is the order of the day rather than creating groups. Secondly, there is no implicit reason to assume that users will have similar behaviour just because they are in the same role or group.

To deal with the second problem, users within the same group or role can have their historical behaviours correlated, to see which users within each group do, in fact, behave in a similar manner. This method compares historical relevance values for content that users in the group have received, with those of the user in question (for the same content). This technique for extending the basic mechanism is essentially collaborative filtering, as described in Section 10.7.2.

It is important not to confuse the group (and other) local relevance and importance values used for determining relevance for content sent to those identities, with their role in ascribing similarity between users.

A resulting statement of similarity from role or group methods might be, “Other users who also work in the development team found this mail to be of high relevance.”

10.7.2 Collaborative filtering

This mechanism uses the behaviour of other recipients of a given piece of content to determine the content’s relevance to the user in question – collaborative relevance. However, rather than just using the global relevance value for the content, the application can correlate the past behaviour of other users to the user in question to improve the relevance predictor.

A key problem with this mechanism is scalability. In the limit, to evaluate the most similar users to the user in question, the system must compare each other users’ actions to those of the user in question. This can generate a three dimensional matrix, showing the user in question, all other users in the system, and then all content relevance values. This would typically be an extremely sparse matrix, because each user would typically receive less than 2% of all mail sent within an organization (average number of recipients of each mail divided by total number of users).

A separate problem would be updating this as a single matrix, which can in part be resolved by recognizing that the matrix can be split back to a series of

independent two-dimensional matrices, one for each user whose relevance is being determined. However, this does not resolve the issue of storing a very sparse matrix that could easily be billions of fields in size. Updating and evaluating these matrices would be computationally onerous to the point of infeasibility in large organizations.

A resulting statement of similarity from collaborative filtering might be, “Other users whose relevance values typically correlate to the user in question (based on comparison for historical content), found this mail to be of high relevance.”

10.7.3 Fingerprinting

Fingerprinting adopts a historical comparison method similar to that of collaborative filtering, but reduces the sparseness of the matrices by focusing purely on a sample of content with large numbers of recipients.

Fingerprinting relies on analysing users’ relevance values for a set of data that has been sent to very large numbers of users, ideally the entire user population. This then provides relevance values for a common set of data, allowing a ‘fingerprint’ of each user’s relevance values to be calculated, and the compared to that of other users. User fingerprint correlation can then be used as a basis for predicting a given user’s relevance values for a new piece of content, based on other similar user’s relevance values for the same piece of content. Thus as more users read a new piece of mail, the initial relevance values for that content for the remaining recipients should be predicted more accurately.

Clearly the challenge with this fingerprinting technique is that there is no strong reason to assume that user’s behaviour for content with small numbers of recipients should directly reflect that for content with large number of recipients. The correlation between users’ behaviour for mails with few recipients and large numbers of recipients will vary from one community to another. However, the feedback mechanisms in the relevance algorithm allow the strength of this correlation to be tested, and the suitability of fingerprinting for use in predicting behaviour for all mail content to be evaluated. Note that for mail content sent with very large numbers of recipients, fingerprinting should usually be a reasonably accurate method for determining similarity.

While this discussion has been focused on users and content, other entities are also susceptible to fingerprinting. The same matrix with content in one dimension can also be applied to groups and domains.

A resulting statement of similarity from fingerprinting might be, “Other users whose relevance values typically correlate to the user in question (based on comparison for a non-representative sample of historical content), found this mail to be of high relevance.”

10.7.4 Cluster models

Cluster models work by segmenting users (and other entities) into categories based on their actions or attributed. In some ways this method is similar to (and partly contains) role and group based similarity, but it does not assume separate relevance and importance values are stored for each cluster. Examples include:

- **Users:** who reply frequently, who forward frequently, who never delete mails, who send mails to large groups, who receive certain mail volumes, who belong to particular groups, who work in particular departments, etc.
- **Jobcodes:** which have the a particular Jobcode parent or type, which are at a certain point in the domain hierarchy, which have a particular retention period, which result in particular external actions, which have certain allowed readers (access rights), etc.

- **Content:** which has a large chain length (number of history submessages), which is of a certain length, which has attachments of certain size or type, which has a large number of recipients, which contains certain keywords, which is linked to a particular Jobcode, which has a particular sensitivity rating, which interacts with certain external systems, etc.

Note that cluster models can potentially be applied to both entities and content in determining similarity and using it as a predictor for user relevance. Content clustering has been included for completion here, though has already been implied in Section 5.3.

A resulting statement of similarity from cluster models might be, “Other users who also forward mails regularly found this mail to be of high relevance.”

10.7.5 Search-based methods

Search-based methods can operate with respect to content, or as a filter to then perform correlative analysis of users’ historical relevance values.

Search-based methods with respect to content treat the similarity problem as a search for similar content in the user’s historical content store. They then compare the relevance values from that user for similar historical content to predict the relevance of the new content. This essentially moves the similarity question to being a function of third party applications, e.g. Autonomy, but still relies on the user’s own actions to determine relevance.

In the second approach, using search-based methods as a filter for comparing other users’ historical behaviour, the application finds similar content to that in question, and then inspects the relevance values given to it by other users. The algorithm then uses these relevance values from other users for similar content as an input to determine the initial relevance of the given content for the user in question.

10.7.6 Frequency analysis

Frequency analysis is another mechanism for reducing data size for collaborative filtering. In this method, each user has a set of other users associated with them based not on similarity, but on frequency. The method works by finding the users who have the most content in common with the user in question, i.e. the most frequent To:, Cc: and potentially Bcc: recipients.

This method does not imply similarity between the sample of say 100 users that have most content in common with the user in question. Rather, it provides a rich (not sparse) matrix of content on which to search for correlations (and anti-correlations) in user relevance values. On a periodic basis, such as each month, the ten most infrequent users in common with a given user can be removed from the matrix, and the ten most frequent users in common (not already identified in the matrix) can be added. This acts to keep the matrix rich with common values.

From an importance perspective, this mechanism can be used to compare a user’s own importance values given to entities such as Jobcodes to those of other users. These correlations can then be used to provide more accurate estimates of initial importance for when a user interacts with that Jobcode for the first time (as opposed to being forced to rely on the global value).

10.7.7 Combinations and detail

In reality, these techniques can be combined for different data types within a user community. Collaborative filtering can be appropriate for small user populations, such as organizations with 1000 or fewer employees. Fingerprinting can be suited to content with large numbers of recipients.

Frequency analysis is generally the standard method for determining similarity in larger organizations.

Mechanisms can also be combined. For example, cluster models can be added as another dimension to frequency analysis. It has already been argued that there is no reason to equally weight users in completely different departments with those in the same departments in calculating collaborative relevance. The same is true for content relating to different processes – the HR Director may be similar to HR users for content relating to HR, but may be more similar to other executives for board minutes etc. Thus adding an additional dimension to comparative similarity can increase the accuracy of the algorithm.

Note that for all these techniques, the contribution of more recent content to the similarity function is greater than that of older content, reflecting the changing nature of the user population as the corporation and their roles within it evolve. An exponential function can be applied to these contributions to decay them over time. Older data is removed from the matrices as new data is added, to ensure that the similarity matrices can be maintained at a given size (rather than escalating into computational infeasibility). The frequency of updating various relevance and importance values can also be determined on a per organization basis, depending on the business requirements and computational power available.

10.7.8 Frequency analysis table

To demonstrate the concepts behind frequency analysis, consider a user, A, and four other users, B, C, D and E, with whom A has a significant volume of content in common. Table 8 lists the relevance values ascribed to historical content, TransMails A through T, that user A has in common with most or all of the other users.

	User				
Content	*User A*	*User B*	*User C*	*User D*	*User E*
TransMail T	5	9	9	4	1
TransMail S	6	2	3	7	4
TransMail R	2	5	7		5
TransMail Q	8		4	6	6
TransMail P	4	7		2	
TransMail O	2	6	9	4	
TransMail N	6	9	1	5	8
TransMail M	9	6	3	8	2
TransMail L	6	3	2	5	5
TransMail K	4	1	4	1	6
TransMail J	7	5		5	6
TransMail I	2	2	3	2	
TransMail H	4	8	6	3	9
TransMail G	3		7		8
TransMail F	1	4	6	3	4
TransMail E	8	7	8	7	2
TransMail D	5	8		7	
TransMail C	7	2	6	6	5
TransMail B	3	9	5	4	8
TransMail A	2	6	8		4

Table 8 – User content matrix

The list of relevance values for historical content allows each user's historical behaviour to be correlated with each other user. The resulting linear correlation values are presented in Table 9.

Correlations					
	User A	*User B*	*User C*	*User D*	*User E*
User A	1	0.042675	-0.35891	0.754378	-0.28248
User B	0.042675	1	0.301382	0.154122	0.122467
User C	-0.35891	0.301382	1	-0.0957	-0.31789
User D	0.754378	0.154122	-0.0957	1	-0.43545
User E	-0.28248	0.122467	-0.31789	-0.43545	1

Table 9 – User correlation matrix

Correlation is a statistical function that indicates the extent of the similarity between two data sets. From Table 9, it can be seen that user A behaves very differently to user B, often has opposite relevance values to user B, and ascribes very similar relevance values to the same content as user D. Thus the behaviour (and resulting relevance value) of user D on a new piece of content sent to both users A and D would be a good indicator of the likely relevance of that content to user A.

In reality, simple linear correlation is not appropriate. User's behaviour can evolve over time, so the correlation formula should be weighted towards the newer content (a non-linear distribution of the square of the error terms in the regression). In addition, there is no reason to assume that the relationship should be linear, thus non-linear correlation techniques may also be appropriate. In a system of very high computational power, the regression formula for each similar pairing could be used to 'forecast' a suitable value for the collaborative component. However, this is overly demanding for most implementations.

10.7.9 Summary

Various techniques have been discussed that can improve the accuracy of collaborative relevance and importance through similarity. Some of these techniques may generate substantial data in matrices for each user, with significant computational overhead in updating relevance values based on similarity. However, it must be remembered that many users spend over three hours each day on email, and given the fully loaded cost of these employees, small increases in productivity can have dramatic ROIs. In the defense and intelligence sectors, the affect can be even more critical – improving relevance of intelligence information can assist the warfighter and help save lives.

10.8 Similarity Implementation Detail

Over to Jon.

10.9 Questions

- The global relevance value for a piece of content – does it only reflect the recipients of that content, given that it could be forward to others, or is it just updated reflecting other users' actions once forwarded?
- Effective relevance – even if you split out a user's own actions and the collaborative aspect of the calculation, you still need to update the user's local number as their local importance values changed based on global importance changes and local importance changes due to the user's own actions on other content.
- Definitions – need to go through and unify definitions so they aren't all different from each other and incoherent.

10.10 Thoughts

We need to compare to Google's page rank technology, and there's a great study of it here: http://www.webworkshop.net/pagerank.html

Other links:

http://ausweb.scu.edu.au/aw2k/papers/lowe/paper.html

Need an expert in this area.

Need to discuss the way BlueSpace does (or will) split up content into a query store and a transactional store.

Calculate relevance for content as it comes in, then change it each time a user does an action, then weight it for people relevant to me, then recalculate importance scores for people on say nightly basis.

May be that due to outliers and random behaviour, if one person leaves a mail open for 30 minutes it's less than 10 times relevant than if ten people do. Need a way to deal with outliers – but not likely to get more than one.

If you forward to your boss and that's more important than one you sent to the office secretary, so you need to run the algorithm iteratively. This is because if the boss is of higher importance the secretary, then forwarding a message to the boss from Bob increases Bob's importance more than if you forwarded it to the secretary.

Importance of content can be done in realtime, importance of people can be reassessed each night.

Taking lag time – so longer you wait before reading, the less important.

Most others deleting mail – might want to weight the users again.

Different people can have different relevance ratings for different respective users – my boss matters more to me than they do to someone else in a different department. So need an n by n matrix.

Each user can affect the relevance scale for any other user as part of a learning cycle / feedback loop.

Big difference compared to normal search engines – each user gets different results.

Relevance scores need to drop off over time – so data that's two years old should have less impact than stuff that's 2 months old.

Relevance actually varies over time – during a particular date period a user may be much more relevant to me due to the project I was working on with them. This clearly reflects search, as relevance could be based on today's relevance values or the relevance value relative to the date the content was written.

Searches should go back into feedback loop – potentially has greater impact on future searches than on messages I'm interested in now, as relevance criteria changed over time. What you're interested in when searching may be very different to what you're interested in when reading – Dilbert cartoons may be the highlight of your day, but you never search back to find particular ones.

Multi-dimensional – if I email Andrew cc Yul and he always keeps it, then email Andrew cc Dan and he always deletes it, then different relevance.

Need to think about common themes in searches – how do you delve into this. Have all the data in a structured way, but how do you recognise themes and related themes through quasi-structured mail content.

Note that in the TransMail view, the TransMail relevance should be some algorithm summing the relevance of the submessages (both current and history) though the relevance components should drop off over time. So the further back in the history you go, the less effect that submessage has on the TransMail's overall relevance.

In order to assist the algorithm limiting the size of data it needs to pull in conducting the search, the administrator could reduce the initial data set. As an example, an administrator can determine that only the last 3 months of data are searched first, and the user is given a button saying "Expand search" to allow them to include older content.

From an algorithm perspective, the application can search first through content with a relevance over a given threshold value, which reduces the computational load and increases response times for building the first pages of search results. Also, typically a user will find what they are looking for in the first couple of pages of results, thus searching through to low relevance content is an unnecessary performance drain.

Note that the time spent reading a given TransMail may also be indicative of relevance, and could be built into the relevance algorithm.

We also need to build in the idea of a federated weightings model, in which corporations provide the derived values for weightings of different actions by web services to a central pool. This increases the sample size, and means that new systems don't start from random values – we build 'understanding' of users' action weightings across corporations. Later, this could also be classified according to different verticals, as health care weightings might be different to manufacturing or military systems.

11 Conclusion

Email is now the most voluminous communication medium on the planet. Business users are drowning in a 'torrent' of content. While in part much of this may be external Spam, much of this email in large corporations is internal Spam through dysfunctional email use and the simplicity of the Cc: function in email applications. A typical business user can receive over two hundred 'real' emails (excluding external Spam) each day.

In addition to ongoing email communication, email has now become the primary source of business records for many legal, regulatory and intelligence investigations. When presented with the volumes of email that come out of these discovery and listening projects, any mechanisms to improve automatic sampling and expansion of the search during the forensic process can provide tangible benefits to the organization carrying out the investigation.

BlueSpace Mail Relevance is designed to alleviate the email pressure on end users – improving corporate efficiency and helping to re-establish 'quality of life' for end users.

BlueSpace Mail Relevance watches how users interact with content and implicitly other users and entities in the system, and takes each user's behaviour as the primary input into the relevance algorithm. The attraction of this approach is that it collects vast amounts of statistical input data to the algorithm, but does not require additional user time. BlueSpace Mail Relevance watches and learns.

BlueSpace Mail Relevance may be applied at a single user level only – based on a user's own actions. It can also be applied across an entire corporation in a collaborative manner, which greatly increases its accuracy by providing more contextual data.

BlueSpace Mail Relevance can also be applied to historical content, providing 'dashboard' summaries and visual representations of communication communities. BlueSpace Mail Relevance can significantly enhance an investigative process, improving sampling and ease of analysis.

The mathematics and computing behind BlueSpace Mail Relevance is certainly intricate and involved, but all of this complexity is hidden from the end user.

BlueSpace Mail Relevance will set a new standard for mail content handling in the global communication marketplace. In a business world in which everything is getting faster all the time, BlueSpace Mail Relevance will ultimately act as the 'swimming coach' and 'flotation aid' to corporate managers around the world.

12 Glossary

This document relies on the following term definitions:

- **Business Records Management (BRM)** – the market in which BlueSpace positions itself, abstracting content from communications that constitute business records, and managing this content in strict compliance with a company's policy.
- **TransMail** – the unit of mail in BlueSpace, and thus the directly analogous 'envelope' to email for users. A TransMail is made of a current submessage (which determines the TransMail subject, sender and recipients), and one or more history submessages.
- **Submessage** – a textual content fragment, authored by a user or computer, referenced by TransMails. Note that a submessage includes or may consist entirely of form components (e.g. check boxes and dropdown selection boxes).
- **History submessage** – a submessage that forms part of the history portion of a TransMail. Any TransMail sent as a reply or forward will have one or more history submessages.
- **Future submessage** – once a TransMail containing a given current submessage, A, has been sent, it may later be forwarded or replied to. Any submessage that becomes the current submessage of one of these later TransMails is a future submessage with respect to submessage A.
- **Attachment** – a file fragment, attached by a user or computer, referenced by TransMails.
- **Population Structure** – consists of all users and describes how those users belong to domains and groups.
- **Entity** – a collective name given to any actor or object in the system, i.e. a user, domain, group, context, Jobcode etc.
- **Domains** – sets of users. They are hierarchical in that domains may have sub-domains. These sub-domains default to having the same properties as the parent (though each property can be specifically changed). Domains are mutually exclusive.
- **Groups** – sets of users that span domains, such that users in different domains may belong to the same group. Groups are not hierarchical and are non-mutually exclusive, i.e. can overlap.
- **Relevance** – an indication of how likely it is that a given piece of content will be of interest to a certain user. Relevance can be applied to both content browsing and content searching, and is the result of an algorithm.
- **Browse Relevance** – a unique value for each piece of content with respect to each user. Browse relevance is what is shown to a user when doing usual tasks such as viewing their inbox.
- **Search Relevance** – search relevance determines ranking of search results for a specific set of criteria input by the user. A given piece of content will have different search relevance values depending on the search criteria as well as the browse relevance values for the content.
- **Importance** – a measure of the level of significance an entity has to another. For example, if John always reads and replies to mails from Sally, then Sally is important to John. Importance is an input to

relevance. Note that a given entity will have lots of importance values, as these values are all relative to other entities.

- **Effective Relevance** – effective relevance is the value shown to a user on screen (either as a browse relevance or search relevance value). Effective relevance can be broken down into two components – single relevance and collaborative relevance.
- **Single Relevance** – the rele
- **Actions** – users' actions determine both relevance and importance values. Both direct user's actions and collaborative actions (actions of other users on the same content) are taken into account.
- **Slider** – a user interface feature that allows a user to view relevance and importance values similar to the bars of a graphic equalizer in a sounds system. The user can click and drag the slider's level to provide feedback to the system, similar to changing the level of a bar on a graphic equalizer.
- **Importance chart** – a visual representation of the way the importance of a given entity has evolved over time.
- **Compliance Policy** – an overarching term defining all aspects of compliance in a BlueSpace system.
- **Rule Set** – a collection of rules that govern the actions that users can carry out in the BlueSpace system.
- **Rule Type** – rule types are categorised as follows: Communication Rules, Readership Rules, Action Rules, Retention Rules and Search Rules.
- **Rule Categories** – divides a Rule Set by rule type.
- **Standard Rule Set** – the Rule Set applied to messages if no Jobcode is specified – in other words, the default rules.
- **Communication Rules** – determines how users, groups and domains can communicate with other users, groups and domains.
- **Readership Rules** – determines which users, groups and domains can read a given piece of content after it has been written during its entire lifecycle (including mail forwards, etc).
- **Action Rules** – determines whether a given piece of content can be forwarded, copied to clipboard, saved, printed, replied to all etc. This Rule Type
- **Retention Rules** – determines for how long a given piece of content should continue to be retained.
- **Search Rules** – determines which content users can view when searching the entire message database.
- **Jobcode** – a way of linking any given submessage with the rules that affect the way that it is handled, allowing exceptions to the default behaviour described by the Standard Rule Set.
- **Compliance Administrator** – a 'super-user' who creates Compliance Policies in the BlueSpace system. Note this is a role (or privilege) given to a user, as are the other administrator roles listed below.
- **Administrator** – a 'super-user' who creates and manages domains, groups and users. Note this is a role given to a user, and a user may have more than one role given to them.

- **Auditor** – a ‘super-user’ who can view a log that indicates every action performed in the BlueSpace system.
- **Compliance Log** – a log that records any event that, if carried out, would contravene the Compliance Policy.
- **Audit Log** – a log that records every significant action that occurs in a BlueSpace system.

---

[i] *Talking Headers*, Katie Hafner and Matthew Lyon, The Washington Post Magazine, 4th August 1996.

[ii] *Legal Lottery*, Melanie Liew, Computerworld, 28th April 2004

[iii] *Talking Headers*, Katie Hafner and Matthew Lyon, The Washington Post Magazine, 4th August 1996.

[iv] *Simple Mail Transfer Protocol*, Johnathan Postel, RFC 821, August 1982

[v] *Is banning e-mail a viable way for IT to avoid being crippled by storage costs?*, Julia Vowler, ComputerWeekly, 16th October 2003

[vi] *Email management too important for CIOs*, Rachel Fielding, Network News, 11th February 2003

[vii] *Legal Lottery*, Melanie Liew, Computerworld, 28th April 2004

[viii] *Relational Messaging White Paper*, BlueSpace Group Ltd, 2nd July 2002

[ix] *BlueSpace Discovery Service*, BlueSpace Group Ltd, 3rd December 2003

[x] *BlueSpace Email Structuring*, BlueSpace Group Ltd, August 2004

[xi] *BlueSpace Integrated Governance Management*, BlueSpace Group Ltd, 15th May 2004

[xii] *BlueSpace Integrated Governance Management*, BlueSpace Group Ltd, 15th May 2004

[xiii] *BlueSpace Email Structuring*, BlueSpace Group Ltd, August 2004

[xiv] *BlueSpace Integrated Governance Management*, BlueSpace Group Ltd, 15th May 2004

[xv] *BlueSpace Integrated Governance Management*, BlueSpace Group Ltd, 15th May 2004

[xvi] *BlueSpace Integrated Governance Management*, BlueSpace Group Ltd, 15th May 2004

EXHIBIT 2073

metadata slip sheet

BegDoc	REV0000087
BegAttach	REV0000087
File Name	BlueSpaceMailRelevancefinal.doc
LastCreateOn	8/4/2005 8:04 AM
LastModifiedDate	8/21/2005 10:24 PM
Custodian	Marston, Justin
Author	.
SortDate	8/21/2005 10:24 PM
Office-Revision Number	81
Last Saved By	Justin Marston
Exhibit No	