

PROVISIONAL APPLICATION FOR PATENT COVER SHEET

This is a request for filing a PROVISIONAL APPLICATION under 37 CFR 1.53(c).

Docket Number: 23682-10519

INVENTOR(s)

Given Name (first and middle [if any])	Family Name or Surname	Residence (City And Either State Or Foreign Country)
Justin P.	Marston	Richmond Surrey, United Kingdom
Andrew Stuart	Hatch	County Durham, United Kingdom

☐ Additional inventors are being named on separately numbered sheets attached hereto.

TITLE OF THE INVENTION (500 characters max.)

MESSAGE CONTEXTS

CORRESPONDENCE ADDRESS

Direct all correspondence to:

☒ Customer Number

00758

ENCLOSED APPLICATION PARTS (check all that apply)

☒ Specification No. of Pages: 93

☒ Return Postcard

☐ Drawing(s) No. of Sheets:

☐ CD(s), Number

☐ Application Data Sheet See 37 CFR 1.76

☐ Other (specify)

METHOD OF PAYMENT (check all that apply)

☒ Applicant claims small entity status. See 37 CFR 1.27

☒ Fee Transmittal Form Enclosed (in duplicate)

☒ Check Enclosed

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.

☒ No.

☐ Yes, the name of the U.S. Government Agency and the Government contract number are:

SIGNATURE OF ATTORNEY OR AGENT

Signature:

Attorney/Reg. No.:

Brian M. Hoffman, Reg. No.: 39,713

Dated:

June 24, 2005

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service pursuant to 37 CFR 1.10 in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below.

Signature:

Typed or Printed Name:

Brian M. Hoffman

Dated:

June 24, 2005

Express Mail Mailing Number:

EV 469682914 US

USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT

META-1018

IPR2025-01248

23682/01000/SF/5145493.1

Effective on 12/08/2004.
Fees pursuant to the Consolidated Appropriations Act. 2005 (H.R. 4818).

FEE TRANSMITTAL **For FY 2005**

☒ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT \$100.00

Complete if Known

Application Number	Not yet known
Filing Date	June 24, 2005
First Named Inventor	Justin Marston
Examiner Name	Not yet known
Art Unit	Not yet known
Attorney Docket No.	23682-10519

METHOD OF PAYMENT (check all that apply)

☒ Check ☐ Credit Card ☐ Money Order ☐ Other (please identify):
☐ Deposit Account Deposit Account Number: 19-2555 Deposit Account Name: Fenwick & West LLP
☒ Charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 ☒ Credit any overpayments

FEE CALCULATION

1. BASIC FILING, SEARCH, AND EXAMINATION FEES

Application Type	FILING FEES		SEARCH FEES		EXAMINATION FEES		Fees Paid (\$)
	Small Entity	Small Entity	Small Entity	Small Entity	Small Entity	Small Entity	
Utility	300	150	500	250	200	100	
Design	200	100	100	50	130	65	
Plant	200	100	300	150	160	80	
Reissue	300	150	500	250	600	300	
Provisional	200	100	0	0	0	0	100.00

2. EXCESS CLAIM FEES

Fee Description	Small Entity	Small Entity
Each claim over 20 or, for Reissues, each claim over 20 and more than in the original patent	50	25
Each independent claim over 3 or, for Reissues, each independent claim more than in the original patent	200	100
Multiple dependent claims	360	180

Total Claims	Extra Claims	Fee (\$)	Fee Paid (\$)	Multiple Dependent Claims	Fee (\$)	Fee Paid (\$)
- 20 or HP =	x	=	0			

HP = highest number of total claims paid for, if greater than 20

Total Claims	Extra Claims	Fee (\$)	Fee Paid (\$)
- 3 or HP =	0	x	= 0

HP = highest number of independent claims paid for, if greater than 3

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper, the application size fee due is \$250 (\$125 for small entity)

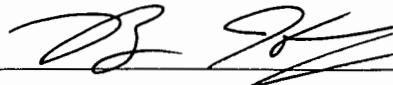
for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets	Extra Sheets	Number of each additional 50 or fraction thereof	Fee (\$)	Fee Paid (\$)
91 - 100 =	0	/ 50 = (round up to a whole number)	x 125	= 0

4. OTHER FEE(S)

	Fees Paid (\$)
Non-English Specification, \$130 fee (no small entity discount)	0
Assignment:	0
Other:	0

SUBMITTED BY

Signature		Registration No. 39,713 (Attorney/Agent)	Telephone (415) 875-2484
Name (Print/Type)	Brian M. Hoffman		Date June 24, 2005

bluespace

BlueSpace Contexts

Contents

1.1 Index

1.1	INDEX.....	2
1.2	LIST OF FIGURES	3
1.3	LIST OF TABLES	3
2	EXECUTIVE SUMMARY.....	4
3	INTRODUCTION.....	5
3.1	EMAIL	5
3.2	EMAIL KNOWLEDGE MANAGEMENT	5
3.3	BLUESPACE.....	6
3.4	BUSINESS BENEFITS	6
4	BLUESPACE TECHNOLOGY.....	7
4.1	BLUESPACE RELATIONAL ARCHITECTURE.....	7
4.2	BLUESPACE EXTERNAL TRANSMail	7
4.3	BLUESPACE GOVERNANCE	8
5	BLUESPACE CONTEXTS.....	10
5.1	INTRODUCTION.....	10
5.2	OVERVIEW	10
5.3	KNOWLEDGE TREE.....	11
5.4	USER EXPERIENCE	11
5.5	ADMINISTRATION AND ACCESS RIGHTS	12
5.5.1	Administrative rights	13
5.5.2	Usage rights	13
5.5.3	Viewing rights.....	14
5.6	CONTEXT PROPERTIES	14
5.7	EXTENSIONS.....	15
5.8	SUMMARY	15
6	EMAIL COMPARISON.....	16
6.1	INTRODUCTION.....	16
6.2	EMAIL UNIT	16
6.3	RELATIONAL RESTRUCTURING.....	16
6.4	EMAIL META-TAGGING	16
6.5	EMAIL INFRASTRUCTURE INTEGRATION.....	17
6.6	SUMMARY	18
7	USER INTERFACE.....	19
7.1	INTRODUCTION.....	19
7.2	BLUESPACE TRANSMail VIEW	19
7.2.1	Mail view	19
7.2.2	Compose / reply / forward view.....	21
7.2.3	Folder view.....	22
7.3	BLUESPACE TRANSMail SEARCH	24
7.4	BROWSING THE KNOWLEDGE TREE	25
7.5	SUMMARY	27
8	BLOGS.....	28
8.1	INTRODUCTION.....	28
8.2	CONTEXTS IN BLOGS.....	28
9	CONCLUSION.....	29

10	GLOSSARY	30
----	----------------	----

1.2 List of Figures

<i>Figure 1 – Email versus TransMail.....</i>	<i>7</i>
<i>Figure 2 – TransMail to external contacts.....</i>	<i>8</i>
<i>Figure 3 – Complete compliance model.....</i>	<i>9</i>
<i>Figure 4 – BlueSpace Contexts overview.....</i>	<i>10</i>
<i>Figure 5 – A simple knowledge tree.....</i>	<i>11</i>
<i>Figure 6 – BlueSpace administrative hierarchy.....</i>	<i>13</i>
<i>Figure 7 – BlueSpace email integration.....</i>	<i>17</i>
<i>Figure 8 – Context in TransMail view page.....</i>	<i>20</i>
<i>Figure 9 – Creating context in TransMail view page.....</i>	<i>21</i>
<i>Figure 10 – Context in TransMail reply page.....</i>	<i>22</i>
<i>Figure 11 – Contexts in TransMail folder page.....</i>	<i>23</i>
<i>Figure 12 – TransMail folder page with context.....</i>	<i>23</i>
<i>Figure 13 – Extended search view with contexts.....</i>	<i>24</i>
<i>Figure 14 – Browsing the knowledge tree.....</i>	<i>26</i>

1.3 List of Tables

<i>Table 1 – TransMail view icons.....</i>	<i>20</i>
<i>Table 2 – Extended view icons.....</i>	<i>25</i>
<i>Table 3 – Knowledge tree interface icons.....</i>	<i>26</i>

2 Executive Summary

E-mail was invented in the early 1970s as part of an academic experiment. According to META Group, 75% of corporate knowledge will reside in email by 2005. Today, business runs on email – it gets used for the vast majority of interactions between people.

Email has catalysed an increase in the speed of business and, as email volume has grown aggressively, the quantity of information presented to employees. 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.

Many systems attempt to help add structure to processes involving user collaboration. Document management systems are an example. But common problem of such systems is their tendency to become a 'ghost town'. As soon as users stop using these applications 'religiously', their value drops off rapidly, as they no longer present a unified view of the process. Email is almost never structured into document management systems due to its sheer volume, but email is now the most common way that users interact and work together.

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.

BlueSpace Contexts allows users to structure their email environment. Content can be placed into a knowledge taxonomy that indicates what the content relates to – such as "Performance", "Sales Targets" and "Sarbanes-Oxley". BlueSpace Contexts is tightly integrated with the users' email interface, so creating a context from a mail view page in the mail client can take five mouse clicks or less, and can be achieved without switching applications or doing a page refresh in an internet interface. This ease of use and simplicity helps to drive user adoption.

BlueSpace Contexts has a 'knowledge tree' – typically a hierarchical directory structure that provides the framework into which content can be linked. A knowledge tree has branches that are 'context categories', such as "Product" or "Sales". These can be further split into subcategories for different processes or divisions. When a user selects a piece of content and decides to add it into the knowledge tree, they create a 'context' – a pointer with meta data and potentially a comment about the content. Context categories act as containers for other context categories and for contexts.

Contexts can be viewed directly from within a user's mail interface, indicating who created it and their reasoning. Contexts can also be browsed like a folder listing through the knowledge tree interface. For example, when a user is writing a specification for a new product, they can browse through contexts created by other users that are in the 'Specification' context category.

This is particularly useful in research and investigative organizations. Consider an example of an investigator who finds an interesting piece of information regarding, say, the tax status of a particular entity in a tax inspection. They can quickly view similar contexts about that corporation, written by other inspectors, by browsing the knowledge tree.

BlueSpace Contexts is extensible to other file types and applications, and can also be integrated with external filtering, indexing and content management applications.

BlueSpace Contexts can empower users to structure content in mail systems without disrupting their normal work experience, allowing true knowledge leveraging of the mail environment and better collaboration with high levels of user adoption.

3 Introduction

3.1 Email

Email was invented in the early 1970s as part of an academic experiment. While its fundamental protocols (such as Simple Mail Transfer Protocol – SMTP) have changed little since their inception in 1982, 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, causing companies' bandwidth requirements to double every six months.

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. 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 Knowledge Management

Email knowledge management can be broadly split into two approaches:

1. **Document Management:** This approach treats email as another part of the content management challenge, and requires users to save emails into documents and then place them into a separate system designed to help manage documents.
2. **Email Search:** This approach attempts to 'crawl' email repositories and then create some type of taxonomic structure that incorporates the content. Alternatively, it may just index the content to allow rapid searching.

Neither of these two approaches has received widespread adoption. The problem with using document management systems to structure email is the sheer volume of email users now send and receive. Even the most ardent document management users, such as lawyers, do not attempt to put the two hundred or more emails they receive each day into a document management system. Sarbanes-Oxley regulation has catalysed the big four audit firms to attempt this approach, but again the lack of integration with the email system ensures that few users rigorously comply with these policies.

There are also several challenges with taxonomic structuring and querying. The first is that most search engines originated on the internet, where from a search engine's perspective, content is available to all or unavailable. It is extremely difficult for search engines to map access rights to unstructured data volumes in complex corporations with matrix reporting hierarchies. The additional challenge for automated taxonomic structuring is the difficulties computers have in determining context. Being 70% accurate is not enough to be useful, and few systems truly approach this level of accuracy. In the future, this may well change, but currently computers cannot 'understand' people.

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.

This white paper describes integrating a content management system directly into an electronic mail environment, empowering users to rapidly associate content with different contexts. This provides the best of both worlds – it allows users to determine contexts, but causes minimal impact on the user experience. Users' decisions on contexts can also feed an analytics engine, act as an input to the BlueSpace Mail Relevance engine and help to determine access rights in the BlueSpace Governance engine.

3.4 Business Benefits

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 specialties, 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:

- Allowing users to browse information in their mail system by context, making finding types of information much easier.
- Facilitating browsing of other users mail stores, from small collaborative teams to large cross-corporate processes.
- Improving search capabilities by providing another criteria for users to select.
- 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.

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.

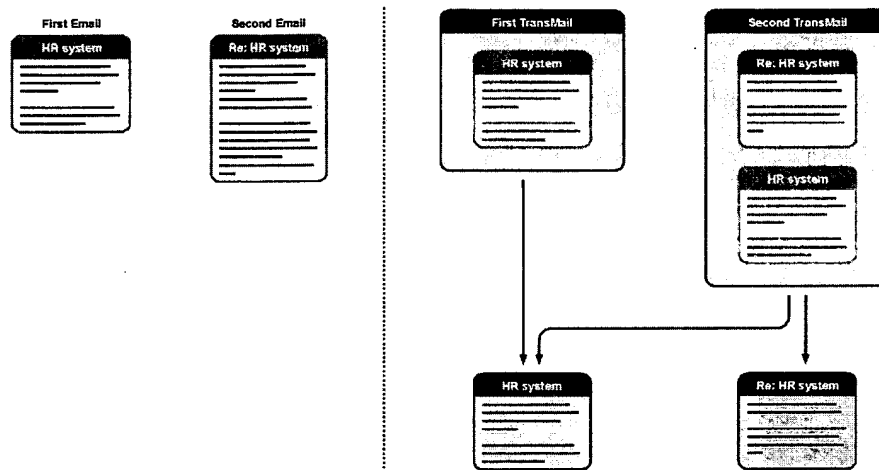


Figure 1 – Email versus TransMail

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. 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 10.

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 BlueSpace External TransMail

BlueSpace can allow TransMail users to send content to external contacts, as shown in Figure 2.

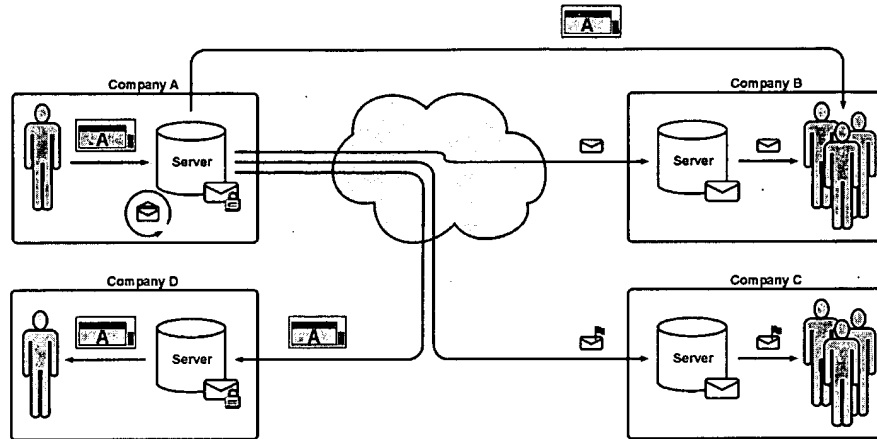


Figure 2 – TransMail to external contacts

Figure 2 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. 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.

4.3 BlueSpace Governance

BlueSpace manages content during its entire lifecycle according to rules laid down in the BlueSpace Integrated Governance Management 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 3.

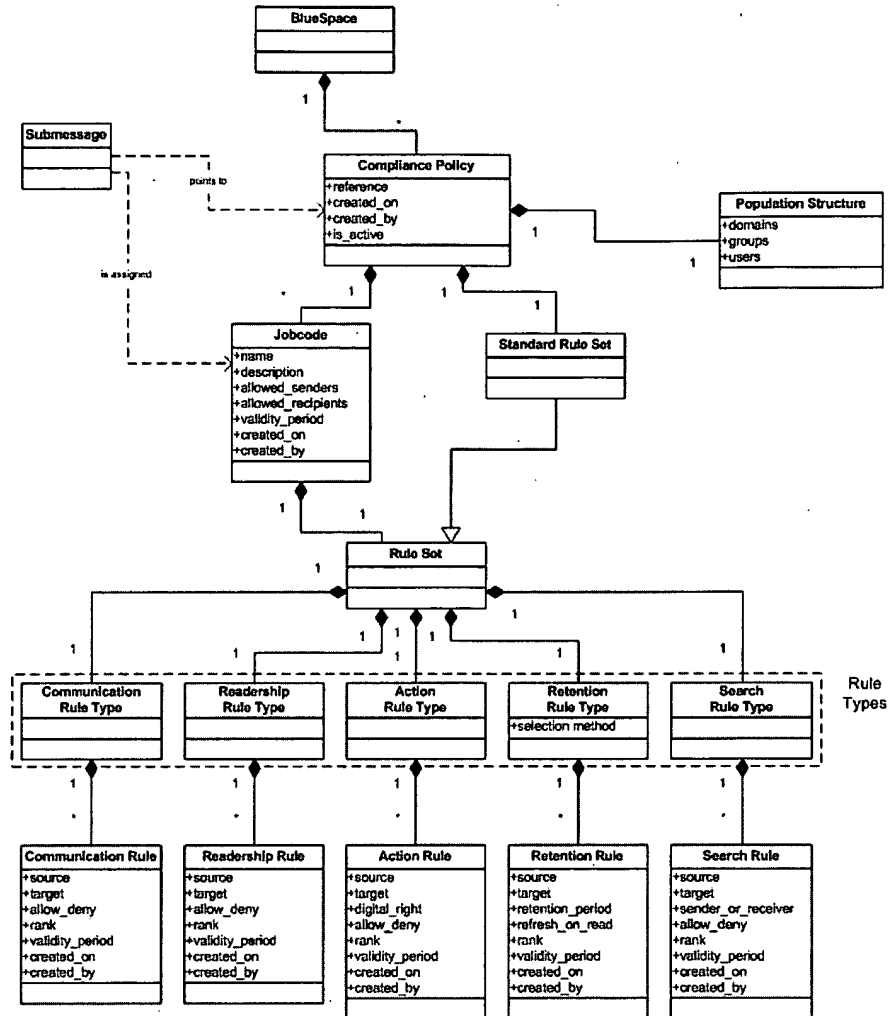


Figure 3 – Complete compliance model

This white paper describes the ability of BlueSpace to layer access rules over a content system, determining who has the right to browse and 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 will not have the ability to search any content that does not pertain to his/her role in the bank.

5 BlueSpace Contexts

5.1 Introduction

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.

Another important aspect of the BlueSpace relational mail model is that each piece of content is stored once, and each further reply or forward references the same piece of content. Any metadata associated with a given submessage can therefore be applied to all instances of that submessage in the mail environment. Ensuring that all metadata regarding a given piece of content is associated with that content over its entire lifecycle in the mail system helps to maximize the value of the metadata.

5.2 Overview

The BlueSpace Contexts model has a structured taxonomy of knowledge to which content can be associated. The examples given in this white paper have a hierarchical tree structure termed a 'knowledge tree', in which each branch point and end is a 'context category'. Context categories can thus have parent categories (the context category above them in the hierarchy) and context subcategories (any context categories that branch off from them in the hierarchy). An association of a context category to a piece of content, along with any associated comment, is termed a 'context'. Note that all terms are defined in a glossary in Section 10.

These contexts can be used by a single user to help them structure their own mail content, but they can also be used collaboratively between users to assist in knowledge sharing. The collaborative aspect coupled to an enterprise-wide knowledge tree adds significant value over contexts being limited purely to each user for personal use.

Figure 4 illustrates the core concepts behind BlueSpace Contexts.

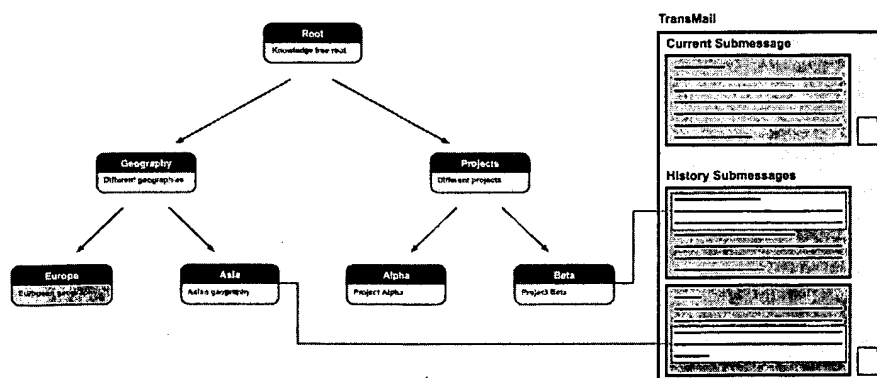


Figure 4 – BlueSpace Contexts overview

In Figure 4, there is a hierarchical knowledge tree represented on the left that has been determined to represent the business context categories in which a given corporation is primarily interested. On the right is a representation of the user interface, showing a TransMail with various pieces of context linked to different contexts. The first part of the first history submessage has been

associated with the context category 'Beta', while the final portion of the second history submessage has been associated with the context category 'Asia'. Each of these two associations has its own context, possibly with a comment made by the creator, that is managed according to the rules assigned to it by the governance policy.

The user interface makes it easy for a user to add contexts, as described in section 7.2.1. It also makes it easy for a user to recognize contexts directly in the email system, rather than having to switch to another application. Contexts may also be used as an input to the BlueSpace search engine. In addition, a user may also choose to browse the knowledge tree itself, viewing content that relates to different context categories as a listing of contexts.

From an enterprise perspective, the knowledge tree consists of both single user and collaborative context categories (populated with contexts). Each context category has a set of associated administrative, usage and access rights that can be mapped to a corporation's own requirements. This can support 'personal contexts', which are created by a user solely to assist their own structuring of their email, through to 'collaborative contexts', which span the whole corporation – or anything between those two extremes.

5.3 Knowledge Tree

A simple knowledge tree is shown in Figure 5.

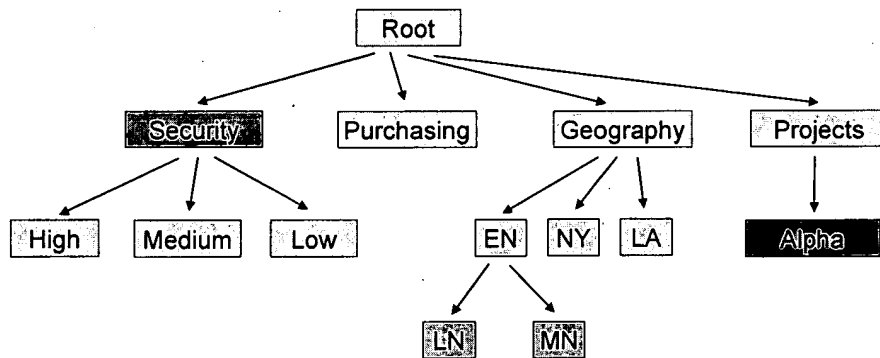


Figure 5 – A simple knowledge tree

In Figure 5, the 'Root' is the highest level of the knowledge tree, which splits into different branches, corresponding to different business contexts that a corporation deals with. 'Security', 'Purchasing', 'Geography' and 'Projects' are all categories of contexts, which themselves can contain more specific context categories – such as different countries, and then different cities within them.

Administration and access rights to knowledge trees and the contexts within them are discussed in more detail in Section 5.5, along with a broader discussion of the workings of the administrative model for BlueSpace TransAct Server.

5.4 User Experience

Figure 4 showed how the BlueSpace Contexts application allows context categories to be associated with different pieces of content in submessages in BlueSpace TransMail by the creation of contexts. This section surveys the use model for users in the BlueSpace Contexts system – the actual user interface itself is illustrated in greater detail, including screenshots, in section 7.

The required user behaviour for creating a context, and thus associating a given piece of content to a context category is as follows:

- The user recognizes that a piece of content in a TransMail is related to a context category they have usage rights to, and they decide to associate the content with that context category.
- The user selects the portion of text with the mouse in the submessage window.
- The user then right clicks the text, and spawns the BlueSpace Contexts pop-out menu.
- This menu follows the hierarchy of the knowledge tree, allowing the user to 'zoom down' to the level they need to find the relevant context category.
- The user clicks the context category to create a context pointing to the content they have selected.
- The application then prompts the user to accept the context category they have selected for the content, and asks the user whether they also wish to incorporate a textual comment into the resulting context.
- The user can input a comment if they wish, and can interact with an additional fields that are determined to be allowed by the governance policy. Examples are specifying potential viewers and editors of the context, if the governance policy allows the context creator to help to determine these rules
- The user then confirms the creation of the context.
- The content is then highlighted in the user interface to indicate it has a context associated with it. The colour of the highlight can be made to correspond to a colour scheme assigned to the different branches of the knowledge tree.
- Rolling the mouse over the highlight then indicates the context assigned to the content, and also shows any comment the user may have added.

This section has surveyed the user experience for regular mail viewing and interaction. A more detailed exposé of the user interface is given in Section 7.2.1, along with the interface for browsing the knowledge tree itself.

Clearly not every user can see a created context – there are access rights regarding which users can view the context.

5.5 Administration And Access Rights

BlueSpace TransAct Server has an administrative model for all entities built around a form of hierarchical directory. An example of such a directory is given in Figure 6.

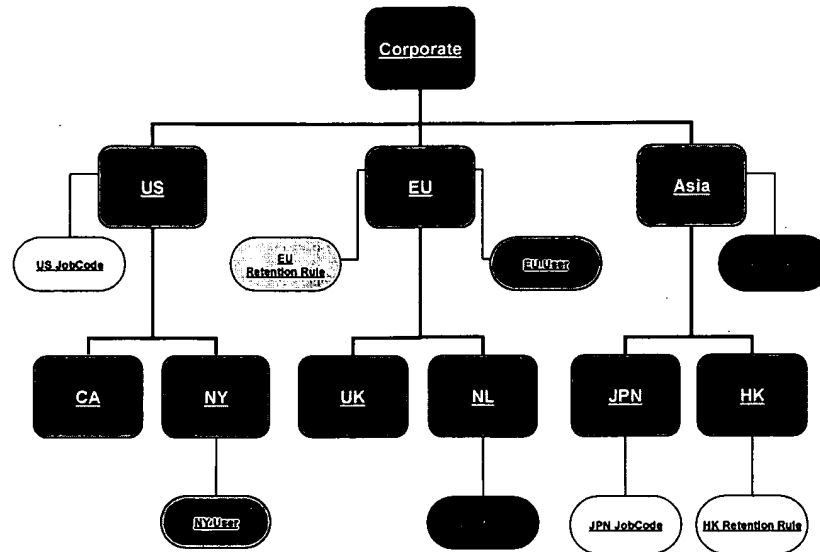


Figure 6 – BlueSpace administrative hierarchy

In the directory shown in Figure 6, the corporate root level branches into different geographies, and various entities of different types are then 'anchored' into this hierarchical tree.

Context categories (and contexts) may be distributed throughout this hierarchical tree, similar to JobCodes shown in Figure 6, or may be managed as a separate branch (added in just below the root level, or at some other point in the tree structure).

Rights over context categories and contexts can be split into the following types:

- Administrative rights
- Usage rights
- Viewing rights

Each of these types shall now be described in greater detail.

5.5.1 Administrative rights

Having administrative rights over a given context category allows a user to change its properties, who can add contexts to the context category, who can view the context category, and also allows creation of context categories underneath that context category in the knowledge tree (i.e. treating the context category in question as a branching point – a container for more context categories).

Administrative rights are not a binary 'all or nothing' right – each 'allowable action' can be made available (or not) by using customized roles. A user might therefore be allowed to create new contexts underneath a given context, but might not be allowed to change the context's retention period.

5.5.2 Usage rights

Having usage rights over a given context category allows a user to associate content to the context category by creating a context. Again this has some additional flexibility through allowable actions. A user may be allowed to add content only they themselves have sent or received to the context category, or may be allowed to add content they have discovered while searching other users' mail stores. They may be allowed to help determine potential viewers

and editors of the context, or these access lists may be determined entirely by the governance policy automatically. Various other limitations can be placed on usage rights to contexts categories, such as times of day that content may be associated with the context category, and the maximum permissible size of a content fragment.

In addition to creation usage rights, a user may also have edit or deletion usage rights. Again, these can be customized extensively using roles and allowable actions.

5.5.3 Viewing rights

Having viewing rights over a given context category or context allows a user to view it. This also has additional flexibility through allowable actions. A user may be allowed to view a given piece of content linked to a context category only when they have received the content directly as part of regular mail flows. Alternatively, a user might be allowed to view the content in the context by browsing the knowledge tree, without having needed to receive the content directly via mail themselves.

Note that technically having viewing rights to a context category is different to having viewing rights to the content associated with it through contexts, but these broadly go hand in hand. For example, a junior HR clerk might have viewing rights to all HR contexts except those that relate to executive management. A greater discussion of these aspects is outside the scope of this white paper, but shall be outlined later in a white paper regarding administration and an advancement of the governance engine.

5.6 Context Properties

Context categories have properties including:

- Their position in the hierarchical tree.
- Any context categories below them (branched from them) in the tree.
- Their validity period, which determines the period of time for which new content may be associated with them.
- Their retention period, which determines the period of time for which they themselves are retained, along with any contexts relating to them (as deletion of a context category enforces deletion of all contexts of that type).
- Various forms of access rights.

Some of these properties are determined specifically for the context category in question. Others can be the result of multiple rules, such as the interaction of retention rules. Contexts have encompass similar properties (although cannot act as containers for more context categories and contexts). In addition, contexts have properties such as the sender, the context author, the content in the submessage that was associated with the context, any relevant comments and timestamps.

Note that retention periods for context categories and contexts do not determine the retention period for the content in the TransMail system – that is determined by the governance rules regarding that content (though contexts may be used as an input to such rules).

Retention periods for contexts determine the period of time for which the association 'pointer' to the content is maintained (along with any comments). When a given piece of content is deleted according to the governance policy, any contexts associated with it are also deleted. When all associations to

content have been deleted (either through the content or contexts expiring) then a context category itself is eligible for deletion.

Note that validity periods, retention periods and other properties are subject to inheritance values that take into account all such values up through the branch structure to the root level.

A final point of note is that the same piece of content can potentially be linked to multiple contexts – this is supported by the application.

5.7 Extensions

The description of the BlueSpace Contexts engine in this white paper has focused on the users' interactions with the mail system. It should be noted, however, that a corporation can decide to extend the context capabilities to the local file system. A user can thus add content in files such as a Microsoft Word document, Excel spreadsheet or PowerPoint presentation to a context category, creating a context that takes its place in the knowledge tree. This functionality employs a plug-in to each file creation application, to facilitate easy creation, management and viewing of contexts. The mechanism of operation is similar to that shown in Figure 7 – the primary difference being that the client application is, say, Microsoft Word rather than Microsoft Outlook.

A second extension specific to files is that of maintaining contexts 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.

Note that the BlueSpace Contexts system can be integrated with external knowledge management tool and their own taxonomies. The administrative model and hierarchy can also be integrated with external directory services, such as LDAP and Active Directory.

The BlueSpace Contexts system can also be integrated with other filtering and indexing technologies from third parties to allow automated creation of contexts. As content passes through the third party application it can either create or 'suggest' contexts. When a corporation decides to allow automated creation, these contexts are treated in exactly the same way as human contexts – and represented similarly in the interface. When a corporation opts for 'suggested' automated context creation, the context is represented differently in the client (using colours and patterns). A user can confirm the context, and it then appears the same in the user interface as human created contexts.

5.8 Summary

BlueSpace assists users in navigating the deluge of mail content they receive each day, by allowing them to structure the content into taxonomies termed knowledge trees. Contexts may be for personal use or for collaboration in anything from small project teams to across the entire corporation. This mechanism also impacts searching. Users can base their searches on conventional inputs, but can also use contexts as criteria if they wish. In addition, users can browse the knowledge tree to find content of a given type.

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 BlueSpace Contexts model 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.

6.2 Email Unit

Pure email unit analysis is the most basic approach to BlueSpace Contexts applied to email content. It allows BlueSpace contexts to be propagated as custom meta-tags that then persist in future versions of the email so long as the user does not substantially change the HTML and/or headers.

This approach provides the least accurate BlueSpace Contexts integration, as it is relatively easy for contexts to become lost when mails are propagated through forwards and replies.

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 approach is less elegant than structuring mail at the front end of the process as in BlueSpace TransMail. It does, however, allow contexts to be propagated into all instances of a given piece of content with greater certainty, and reduces the system's reliance on the settings of the email propagation process.

6.4 Email Meta-Tagging

The relational restructuring process has two phases to its operation:

1. **Upstream Structuring:** Tagging outgoing email by various means to facilitate content recognition when it is sent back to the corporation as part of a different email.
2. **Downstream Analysis:** Taking incoming email content through a digestion engine that resolves content fragments (submessages) that have already been stored by the application, and then rebuilding the relational links.

The first method – upstream structuring – has analogy in the application of BlueSpace Contexts to regular email outside of the BlueSpace TransMail system. The BlueSpace Knowledge Server (revisited in Section 6.5) has the capability to embed contexts within emails, using both tags within the body of HTML emails and email headers (akin to meta-tags) in both HTML and text emails.

6.5 Email Infrastructure Integration

BlueSpace Contexts can be integrated with the email infrastructure in two ways, which are not mutually exclusive:

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

Clearly the client-side integration allows contexts to be created rapidly and directly in the email user interface. The server-side integration assists in recognizing existing content and 'reinserting' any lost contexts that were corrupted by other email servers.

These two methods are illustrated pictorially in Figure 7.

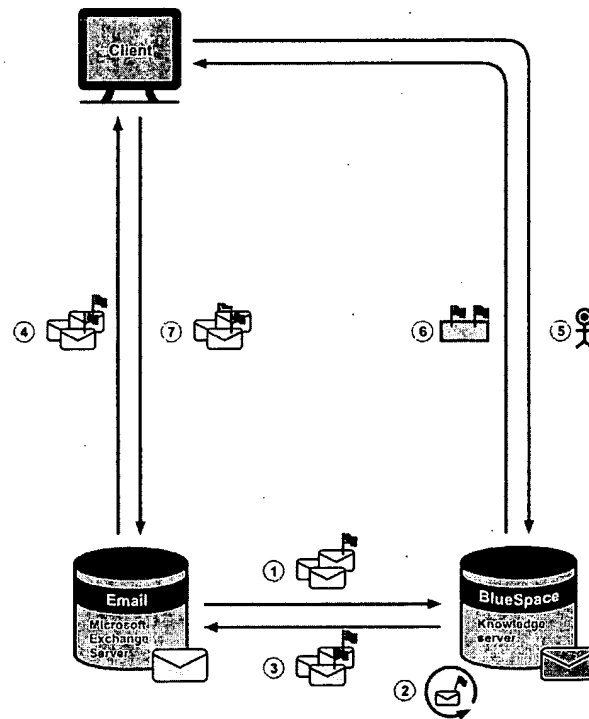


Figure 7 – BlueSpace email integration

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 analyses the incoming emails, and reinserts any missing contexts based on its own store.
3. The emails complete with contexts (represented by BlueSpace flags) are then replaced on the email server.

4. The email client, e.g. Microsoft Outlook, downloads the emails complete with their accompanying contexts.
5. The user can add new contexts (and interact in other ways with contexts), and these actions are sent to the BlueSpace Knowledge Server.
6. The BlueSpace Knowledge Server returns newly created contexts to the plug-in, so that these can be represented in the email client.
7. One of the actions carried out by the end user is to send emails in replies and forwards – these include the contexts positioned by the user and the BlueSpace Knowledge Server.

Note that steps 5 and 6 happen continually, in that new contexts will be created regularly in a large company, and the user can be continually adding new contexts themselves. All the steps in the process are not shown for the sake of brevity, but the knowledge tree is maintained on the server and dynamically synchronized with the plug-ins in the email clients.

Various other views and interfaces analogous to those presented in Section 7 are also present for BlueSpace Contexts for email, though again are not presented here for the sake of brevity.

6.6 Summary

BlueSpace Contexts can be applied to an email environment, and used as a standalone knowledge management system for regular email systems with no TransMail penetration.

Note that to apply the BlueSpace Contexts model to email, the system does not need to restructure the content into the BlueSpace relational store. With client-side integration, BlueSpace can empower the user to create contexts at the client level rather than relying on inspecting mail data. With server-side email infrastructure integration, BlueSpace can recognize context text fragments in emails – along with meta-tags and actors (author and recipients) – and use these as the basis for reinserting lost contexts.

7 User Interface

7.1 Introduction

As first presented in Section **Error! Reference source not found.**, BlueSpace Contexts in mail cause a significant improvement in the end user experience of a mail environment. BlueSpace Contexts empowers users to link content in TransMails (or regular emails) to context categories that describe the type of information contained in the content.

This section is dedicated to describing the user interface for BlueSpace Contexts, and more specifically, the BlueSpace Contexts functionality applied to TransMail 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 input from BlueSpace Contexts:

- Mail view – when viewing a single TransMail.
- Compose / reply / forward view – when creating a TransMail.
- Folder view – a folder listing showing many TransMails.

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.

An example of the mail view incorporating BlueSpace Contexts is shown in Figure 8.

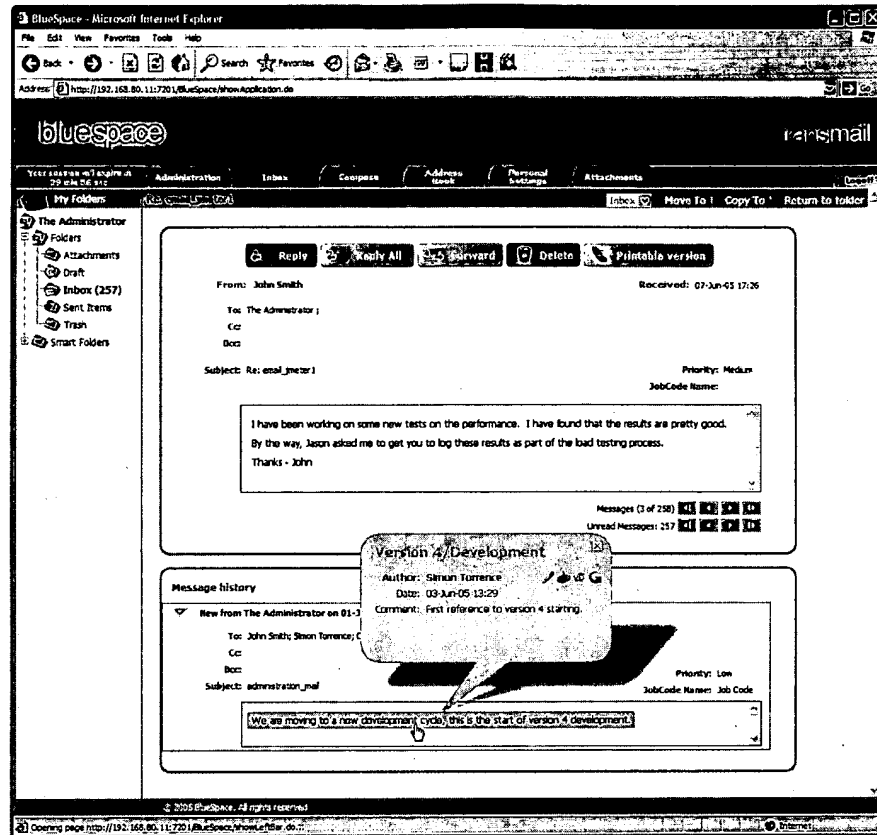


Figure 8 – Context in TransMail view page

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

Icon	ALT Text	Action
	Edit context	Spawns a popup that allows the user to edit the context.
	Show history	Spawns a popup presenting the history of users who changed the context and their respective comments.
	Context visualization	Spawns a popup offering various visualizations of the context and its interaction with the knowledge tree.
	Go to knowledge tree	Shows the knowledge tree interface with the context in question as the focus item.

Table 1 – TransMail view icons

Figure 8 shows a context pop up that indicates the context's context category, the author, the date it was created and a comment by the author. The four icons and their purposes are given in Table 1. The ability to edit the context and show the history is only present if the icon is not greyed out, indicating that the user has the right to access the functionality.

A user can also create a context by selecting text, and then right clicking the selection to spawn the 'Create Context' popup, as shown in Figure 9.

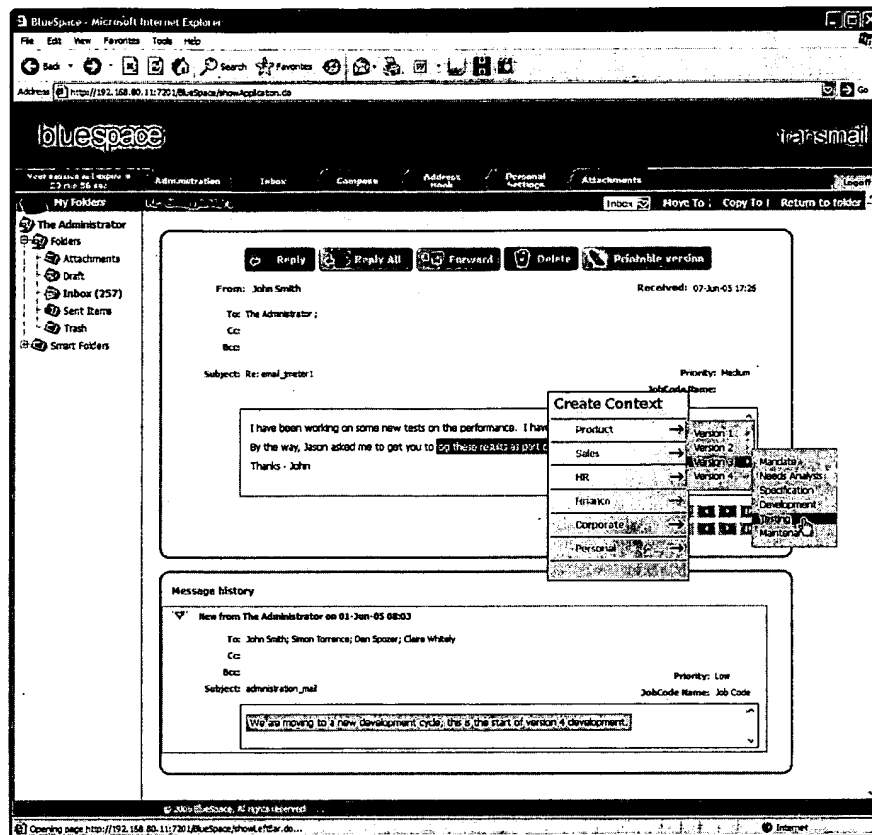


Figure 9 – Creating context in TransMail view page

When creating a context, a user must specify the context category in the knowledge tree, which is presented as a pop out menu.

7.2.2 Compose / reply / forward view

The compose, reply and forward view pages provide very similar interaction to the mail view page, allowing the user to view contexts that have been added to history portions, and also to add contexts into a TransMail at the time of creation. An example of this is shown in Figure 10, in which a user (Jane) decides to link a piece of content regarding improved performance into the Sales branch of the knowledge tree. This allows her to ensure that sales gets access to the content, even if the user (John Smith) she is communicating with forgets to tell sales directly.

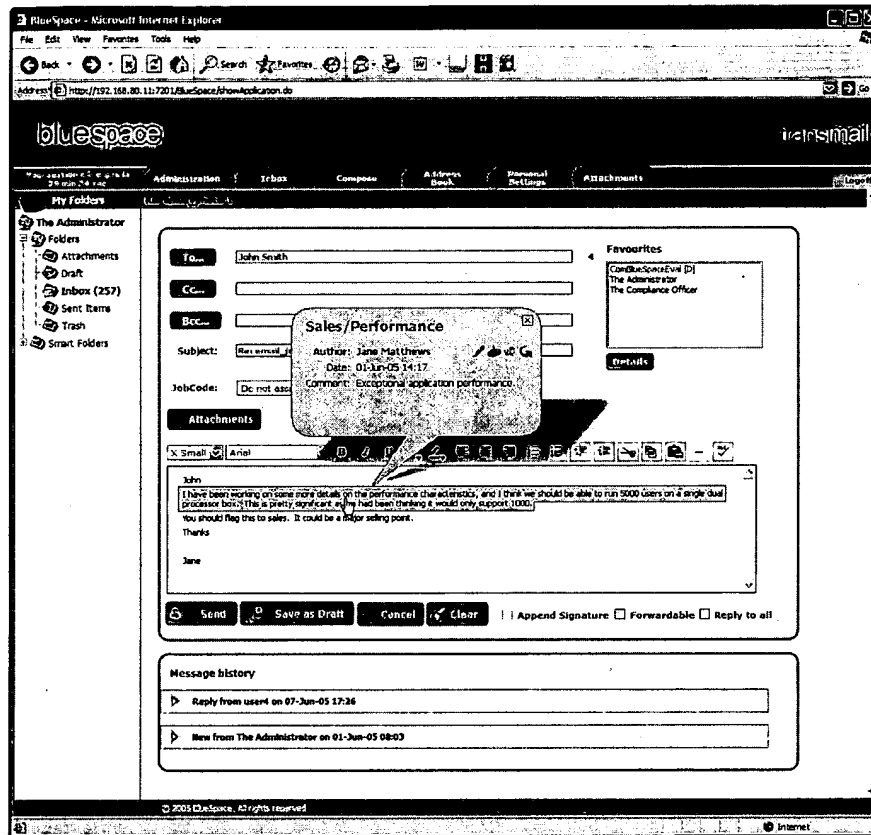


Figure 10 – Context in TransMail reply page

7.2.3 Folder view

Contexts are represented in the TransMail folder view by a column, with colours indicating to which branch of the knowledge tree the context belongs. This is illustrated in Figure 11.

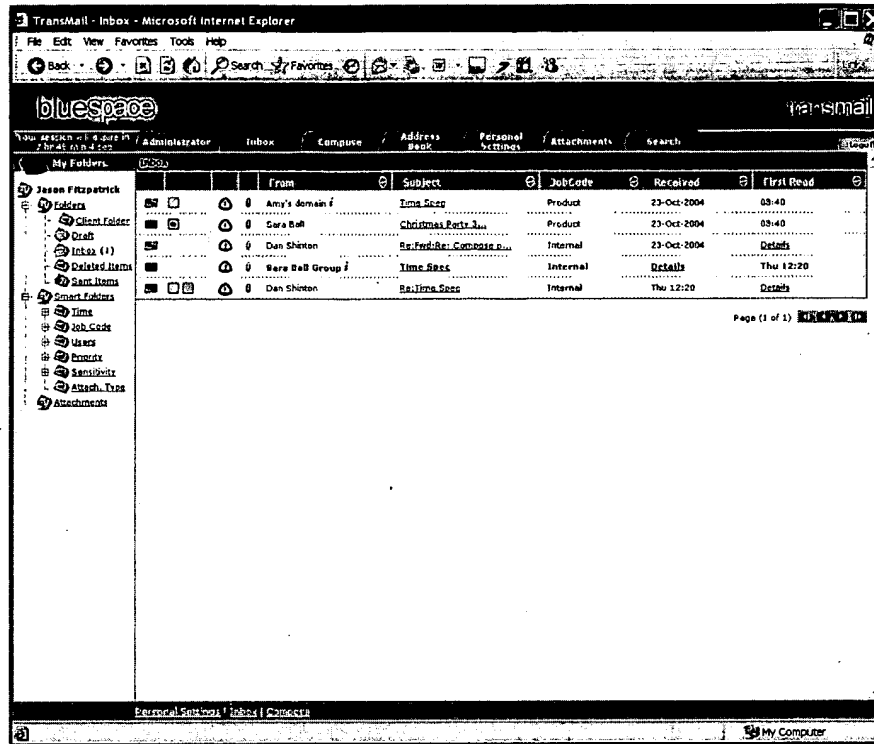


Figure 11 – Contexts in TransMail folder page

When a user clicks one of the context squares, they are shown the corresponding context as a popup (they do not need to view the TransMail to access the context). This is illustrated in Figure 12.

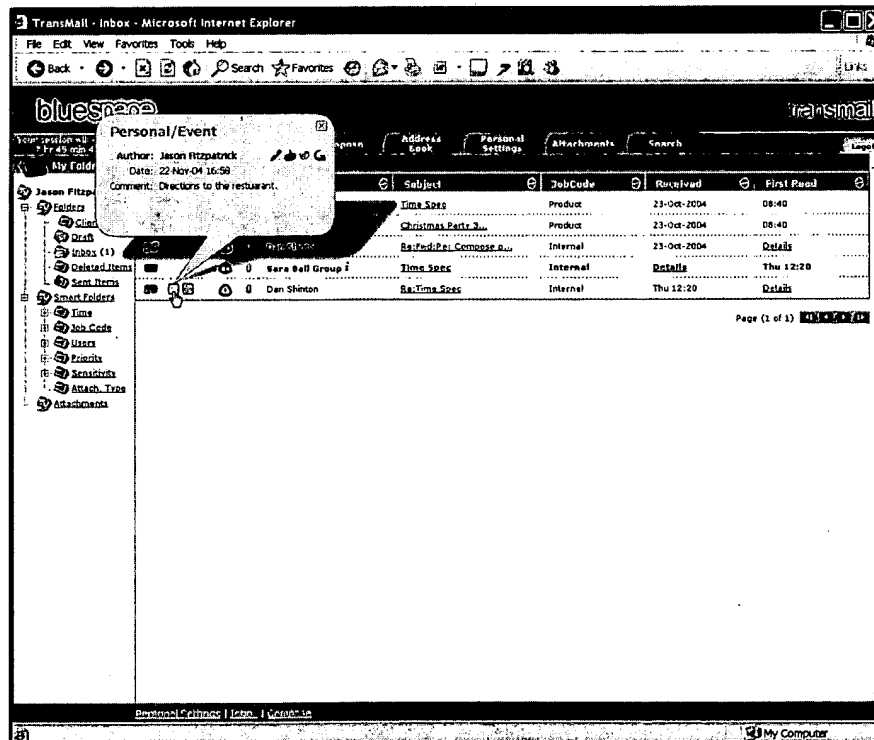


Figure 12 – TransMail folder page with context

7.3 BlueSpace TransMail Search

From a contexts perspective, contexts are a potential user input into the search algorithm, and are also presented in the search results.

Search results in BlueSpace have two forms, a folder view and an extended view. The folder view is broadly similar to that shown in Figure 11. The extended view is very different, and has more in common with search engine results pages. An extended view of search results incorporating contexts is shown in Figure 13.

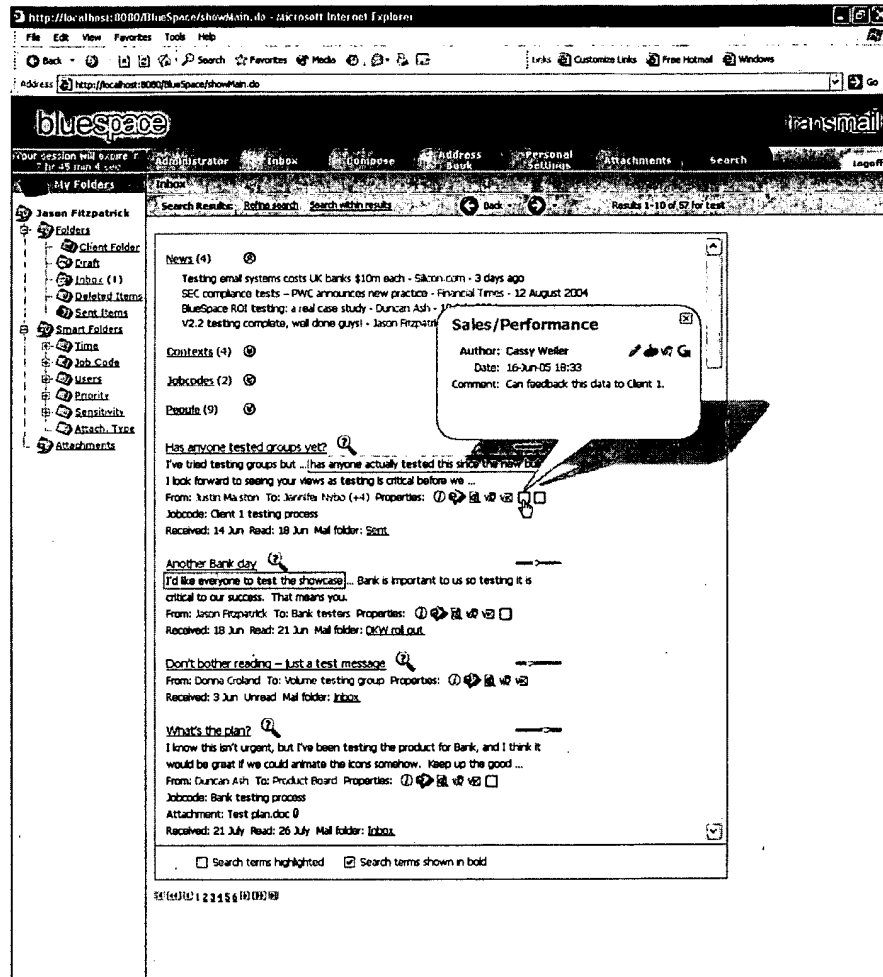


Figure 13 – Extended search view with contexts

The additional icons shown in Figure 13 are described in Table 2.

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 a further popup.
	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, 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.
	Visualise mail	Shows a popup with a visualisation of the mail interaction that led to this TransMail.

Table 2 – Extended view icons

A full description of Figure 13 is given in the BlueSpace Mail Relevance white paper. The point of interest for this white paper is that contexts are shown as part of the search results, and text in the message bodies that is included in contexts shows highlighting as in the mail view page illustrated in Figure 8.

7.4 Browsing The Knowledge Tree

The interface for browsing the knowledge tree is intended to have similar resemblance to conventional email applications, so that it is intuitive for end users. The interface is illustrated in Figure 14.

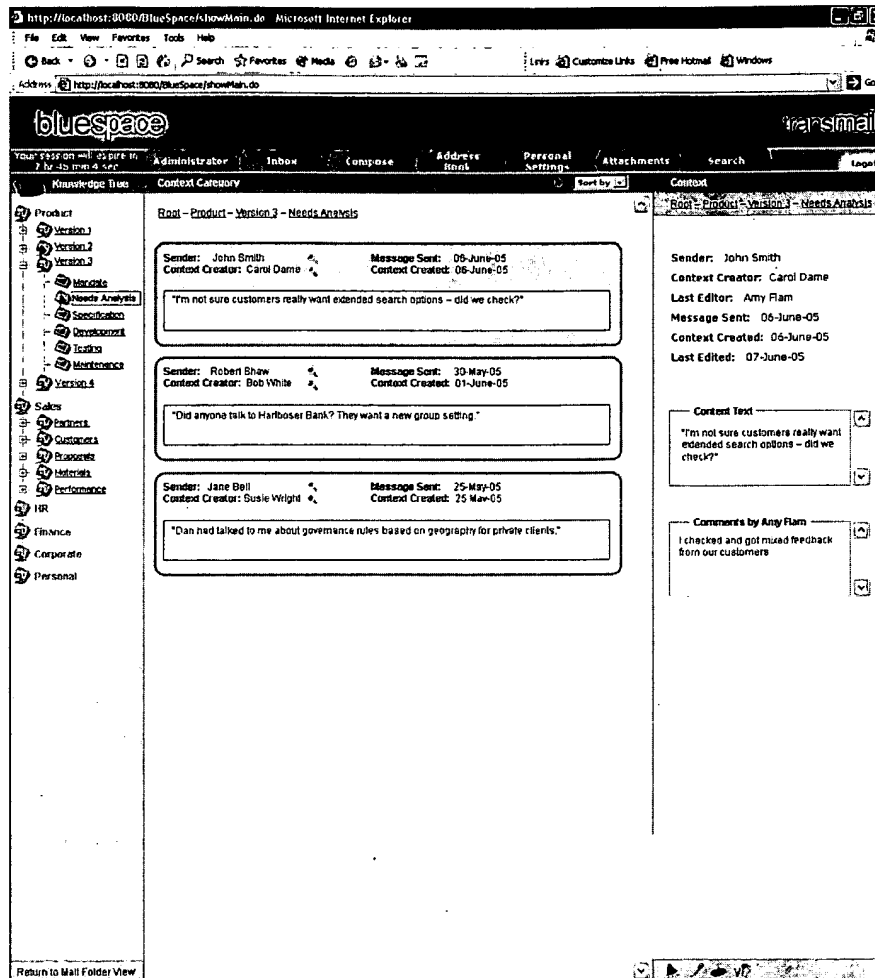


Figure 14 – Browsing the knowledge tree

The additional icons shown in Figure 14 are described in Table 2.

Icon	ALT Text	Action
	Subscribe to this [entity]	This icon allows a user to subscribe to a context category or context, so that they are alerted to any additions or changes to the entity.
	Show more [type] from this user	Presents a custom folder view showing all contexts in that context category that have content written by that user, or were created as a context by that user (depending on the what field the icon is next to).

Table 3 – Knowledge tree interface icons

The browsing interface allows the user to view content in the TransMail store as a structured knowledge repository. Browsing by category makes it much easier to find information related to a certain subject. Consider a needs analysis process for a new version of the product, e.g. version 4, building on the scenario illustrated in Figure 14. A business analyst can browse through all the Needs Analysis contexts from version 3 as a starting point for the new version. They can look to include features that people talked about but didn't make it into the previous version. They can also browse contexts for the

Needs Analysis for version 4, that users have created in the run up to the new version.

Contexts can break down conventional walls between departments – if people in the sales department want to make sure their ideas get to those in development, they can specify contexts for the Needs Analysis category. This is far better than emailing them, for the following reasons:

- They will probably email their idea to the wrong person.
- That person will likely never get round to forwarding it.
- If the identity of the person doing the Needs Analysis hasn't been decided on yet, then they can't email that person at the time.
- Even if they email the right person, it's highly likely that by the time that individual has to do the next needs analysis process, they will have lost that sales requirement in their inbox.

Overall, this is more of a 'smart pull' publishing approach as opposed to a 'push and hope' approach, yet it is still tightly integrated with the electronic mail (push) environment.

A user can sort the context listing for a context category by the following criteria (all either ascending or descending):

- Sender
- Original context author
- Most recent context editor
- Date – of either content or context
- Relevance – of either content or context, as defined in the BlueSpace Mail Relevance white paper
- Type – of either content or context, based on a third party application's structuring of contexts within a given context category (so that all contexts related to 'directories' are shown together, followed by those mentioning 'SMTP').

The ability to subscribe to context categories and to contexts allows a user to keep abreast of new content in an area they are interested in, which previously would have remained buried in the mail environment without them ever finding out about it. In this way, they can leverage knowledge in the mail system rather than being accidentally excluded – hardly a rare occurrence in rapidly evolving corporations with frequent change processes and staff turnover.

7.5 Summary

This section has overviewed the BlueSpace Contexts interface with respect to the end user – how they see it. It contains a number of features that improve the user's experience. By empowering users to add structure to mail content in real-time, 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 Contexts In Blogs

While this white paper has focused on applying the BlueSpace Contexts system to a mail environment, the same system can be applied to blogs. Integration with the blogs interface allows a user to select content and create a context that relates to the content. This is related to Section 5.7 – it is just another medium to which the contexts functionality can be extended.

BlueSpace contexts 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 6.5 and illustrated in Figure 7.

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

9 Conclusion

Email is now the most voluminous communication medium on the planet. Business users are drowning in a 'torrent' of content.

BlueSpace Contexts is designed to help to alleviate the email pressure on end users – improving corporate efficiency and helping to re-establish 'quality of life' for end users. It facilitates better collaboration by improving knowledge sharing, and coupling the functionality into the email user experience.

BlueSpace Contexts empowers users to add content into a knowledge tree without leaving the email or TransMail application. By massively reducing the user requirement, in terms of numbers of clicks and amount of disruption, BlueSpace Contexts can achieve greater user uptake than conventional approaches to enterprise content management.

BlueSpace Contexts may be applied at a single user level only – allowing a user to manage their own information world in isolation. It can also be applied across an entire corporation in a collaborative manner, which greatly increases its use by enabling teams to share their structuring efforts and results.

BlueSpace Contexts 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 Contexts can provide tangible business advantage, particularly in communities that engage in substantial research and investigative activities.

10 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.
- **TransAct Server** – the BlueSpace J2EE server application that acts as a platform for TransMail and other services and communication tools.
- **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.
- **Knowledge Tree** – a taxonomic structure that can be employed to order contexts into a sensible arrangement.
- **Context Category** – a container for contexts and other context categories in the hierarchical knowledge tree structure.
- **Context** – an association made between a context category and a piece of mail content, including a reference to the specific piece of content and any comment by the creator of the context. This context can be associated with the specified content over its entire lifecycle, subject to retention rules.
- **Personal Context** – a context created for the sole use and viewing of a single user, to help them manage their own email stores.

- **Published Context** – a context created for the sole use of a single user, but that may be viewed by multiple users.
- **Collaborative Context** – a context shared either in both usage and viewing by multiple users.
- **User Roles** – can be assigned to users to help to determine which actions they can carry out, determining rights over aspects such as administration.
- **Allowable Actions** – each action a user can perform in the BlueSpace application, which may be correlated with customized user roles for greater flexibility.
- **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.
- **Validity Period** – the period of time for which a given entity can be an actor in a BlueSpace system, e.g. the period of time for which a context category can be used by users.
- **Retention Period** – the period of time for which a given piece of content or entity is retained before being rigorously deleted.
- **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' (role) 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' (role) 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' (role) 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.



BlueSpace Email Structuring

Contents

1.1 Index

1.1	INDEX.....	2
1.2	LIST OF FIGURES	3
1.3	LIST OF TABLES	3
2	INTRODUCTION	4
2.1	EMAIL	4
2.2	BLUESPACE.....	4
2.3	BLUESPACE EXTERNAL TRANSMAIL	5
2.4	BLUESPACE EMAIL ANALYSIS	6
2.5	BLUESPACE ENGINE.....	7
2.6	DOWNSTREAM AND UPSTREAM	7
2.7	SUMMARY	7
3	LEGACY MIGRATION	8
3.1	PURPOSE	8
3.2	DOWNSTREAM ANALYSIS	8
3.3	SUMMARY	9
4	EMAIL SEARCH.....	10
4.1	PURPOSE	10
4.2	SEARCH RIGHTS	10
4.3	DOWNSTREAM AND UPSTREAM.....	10
4.4	DOWNSTREAM ANALYSIS	10
4.5	UPSTREAM STRUCTURING	11
4.5.1	Single gateway.....	12
4.5.2	Multiple gateways.....	13
4.5.3	Email server plug-in.....	13
4.6	SEARCH CRITERIA	14
4.7	SEARCH RELEVANCE.....	14
5	SMTP GATEWAY.....	15
5.1	INTRODUCTION.....	15
5.2	WHY BOTHER?.....	15
5.3	NATURE OF INTERACTION	16
5.4	CONTENT FLOW	16
5.5	AUTHOR ANALYSIS	17
5.6	SUMMARY	18
6	DOWNSTREAM ANALYSIS.....	19
6.1	INTRODUCTION.....	19
6.2	EMAIL CONSTRUCTION, STORAGE AND FORMATTING.....	19
6.2.1	Construction.....	19
6.2.2	Email Body Structure	20
6.3	CONVERSION	21
6.3.1	Stores.....	21
6.3.2	Procedure	22
6.3.3	Procedures and Functions.....	23
7	UPSTREAM STRUCTURING	24
7.1	TAGGING.....	24
7.1.1	Plaintext Tagging	24
7.1.2	HTML Tagging.....	27

7.2	RECOGNITION AND CONSOLIDATION.....	28
7.2.1	Recognition Process.....	28
7.2.2	Consolidation Process.....	28

1.2 List of Figures

Figure 1 – Email versus TransMail.....	4
Figure 2 – TransMail to external contacts.....	5
Figure 3 – BlueSpace email analysis.....	6
Figure 4 – Legacy migration.....	9
Figure 5 – BlueSpace email search.....	11
Figure 6 – Single gateway.....	12
Figure 7 – Multiple gateways.....	13
Figure 8 – Email server plug-in.....	13
Figure 9 – BlueSpace SMTP gateway.....	16
Figure 10 – Reply with separate new content.....	29
Figure 11 – Reply with interspersed new content.....	30

1.3 List of Tables

Table 1 – BlueSpace structuring HTML tags.....	28
--	----

2 Introduction

2.1 Email

Email was invented in the early 1970s as part of an academic experiment. Whilst its fundamental protocols (such as Simple Mail Transfer Protocol – SMTP) have changed little since their inception in 1982, 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, causing companies' bandwidth requirements to double every six months.

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. But there is no way to mine this data effectively. 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. Again, this reflects the history of email as a person-to-person communications system, rather than a system designed for knowledge management.

2.2 BlueSpace

BlueSpace has developed a messaging platform that is different from existing messaging and mail systems. As introduced in Section 2.1, email photocopies content with each generation of reply and forward, as shown in Figure 1.

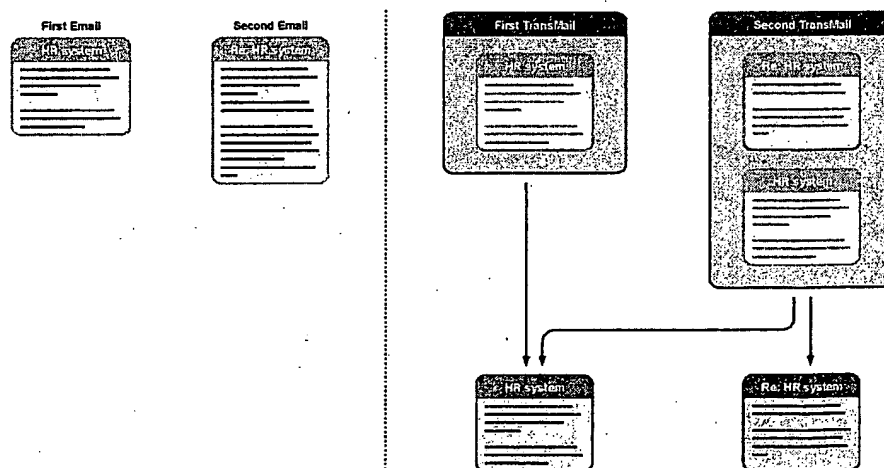


Figure 1 – Email versus TransMail

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. Both red and green messages are termed 'submessages' in the BlueSpace environment. 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.

2.3 BlueSpace External TransMail

BlueSpace can allow TransMail users to send content to external contacts, as shown in Figure 2. Companies A and D have BlueSpace TransMail servers, whilst companies B and C have regular email servers.

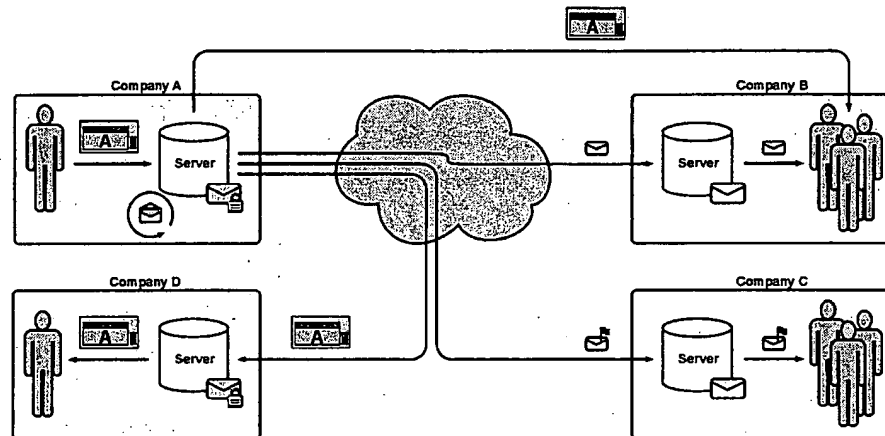


Figure 2 – TransMail to external contacts

Figure 2 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.
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. 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. Interpreting any replies to sent messages can be a challenge, especially in corporations with BlueSpace systems deployed alongside regular email systems.

This white paper focuses on structuring email into the relational system that BlueSpace has deployed for its own unit of messaging, TransMail. To do this, BlueSpace analyses unstructured email content, and pieces back together the implicit relational pointers that exist in email content. BlueSpace can also use intelligent tags to identify content sent outside the enterprise via email to assist

in recognising those content fragments when they return to the enterprise as part of another email.

2.4 BlueSpace Email Analysis

There are two types of email that BlueSpace must analyse to place into a relational structure – historical email and real-time email (i.e. ongoing conversations). In Figure 3, these two types are placed in the context of the three purposes for which BlueSpace analyses incoming email.

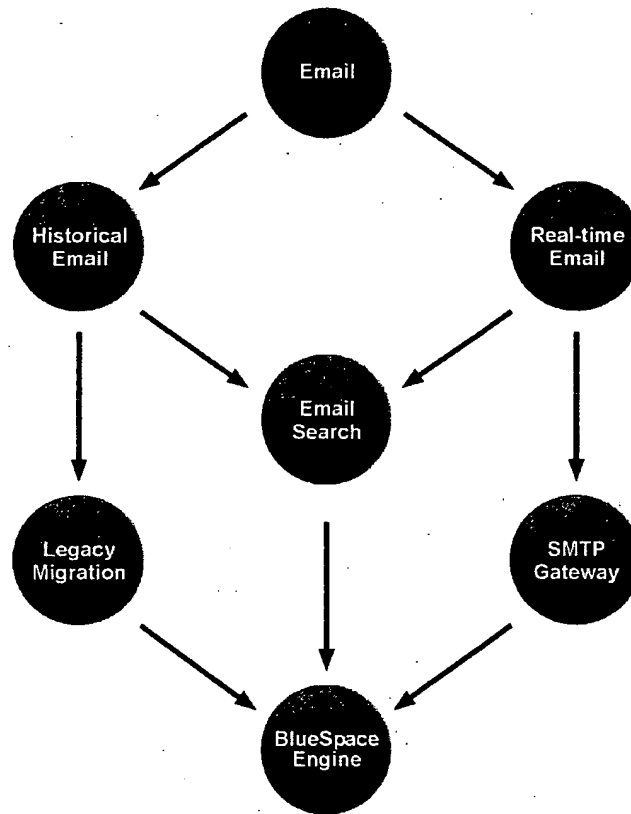


Figure 3 – BlueSpace email analysis

BlueSpace analyses email content as part of three different processes:

1. **Legacy migration:** In BlueSpace deployments in which a corporation is replacing a regular email system with a BlueSpace mail system, there is a significant volume of existing email that users still need. This is a legacy migration, typical of many software installations.
2. **Email search:** BlueSpace has the capability to sit behind existing email systems, and act as a knowledge management repository for email. Once deployed, new email being sent by users must be copied into this search archive to keep the knowledge store up to date.
3. **SMTP gateway:** In many BlueSpace deployments, interaction with external contacts by regular email is a requirement. In order to understand how new content might be best integrated into the BlueSpace relational architecture, a BlueSpace system must analyse the mail it receives so as to not duplicate content in the mail store, but rather use pointers instead.

The BlueSpace engine is discussed in detail in Section 2.5.

2.5 BlueSpace Engine

BlueSpace has an intelligent rules engine in its content management system. This rules engine controls what happens to each piece of content in the TransMail environment. Aspects that are controlled include direct user communication rights, access rights to content that has already been written, action rights (e.g. forwarding and printing content) and retention periods at the end of which content is rigorously deleted. The relational architecture of the BlueSpace TransMail environment allows far greater control than regular email systems. This is because BlueSpace can maintain a single instance of each piece of content, rather than 'photocopying' many copies of the same piece of content across the entire system with no ability to audit the copies.

2.6 Downstream and Upstream

There are at least two types of data structuring that BlueSpace can perform on email content:

1. **Downstream:** Downstream analysis is work carried out on email content after it has been written.
2. **Upstream:** Upstream structuring involves modifying emails as they are created in order to generate structure.

In downstream analysis, a piece of software inspects all email received by the corporation, and then places the content into a store designed for querying. Prior art systems never attempt to introduce structure at the message level. Rather, they inspect emails (that contain multiple messages and message fragments) and ensure they do not keep multiple copies of the same email (but do keep multiple copies of the same messages in different emails).

Upstream structuring can greatly facilitate downstream analysis. A principle in BlueSpace is to maintain context of communications. Fitting an external reply or forward into a corporation's relational messaging store can be a bit like trying to 'find a needle in a haystack'. How does the corporation check its entire message store to see if any component of an incoming email already exists in the message store? Upstream structuring involves tagging each message fragment with a unique reference number, typically hidden from the external recipient. When an external user replies to an email they have received with an embedded unique reference number, the BlueSpace application can find the embedded reference number. This reference number allows the BlueSpace application to recognise the content fragment in the email, and then use a relational pointer to the existing instance of the content fragment rather than duplicating it.

2.7 Summary

Email has a duplicate nature that prevents rigorous management of content and hampers its use a knowledge management system. BlueSpace has a unique relational messaging architecture that facilitates knowledge management. This white paper addresses mechanisms to convert incoming email from external sources into the same relational architecture used by BlueSpace TransMail. This white paper shall now focus on the three processes – legacy migration, email search and the BlueSpace SMTP gateway. In discussing these three processes, it shall also outline mechanisms for downstream analysis and upstream structuring.

3 Legacy Migration

3.1 Purpose

A purpose of BlueSpace legacy migration is to allow corporations to turn off their old email systems when moving to a BlueSpace environment. This has significant benefits:

- Corporations do not need to run two messaging systems next to each other, and bear the cost associated with this.
- Corporations get immediate cost savings from mail centralisation compared to running distributed servers that need managing.
- Converting regular email content to a relational structure provides additional immediate cost savings through storage reduction. This figure is typically over 50%, even for relatively centralised environments with archiving products such as KVS.
- Existing back up tape repositories can be parsed, the full email system contained across all of them rebuilt, and the back up tapes then redeployed or disposed of. This brings all historical content online for management.
- The same governance model used by BlueSpace to manage the TransMail environment can be applied to historical content, particularly in respect to retention of mail content by user, but also (if required) with respect to allowing users to search particular types of content. Access rights are mapped over search content as outlined in Section 4.2.

3.2 Downstream Analysis

In order to migrate legacy content into a BlueSpace relational architecture, the BlueSpace application attempts to 'piece back together' the context of the original communications. As shown in Figure 3, legacy migration occurs purely on historical content, i.e. content that users have already written. Given this, only downstream analysis techniques are available, as described in Section 2.6.

The required result from legacy analysis is represented graphically in Figure 4.

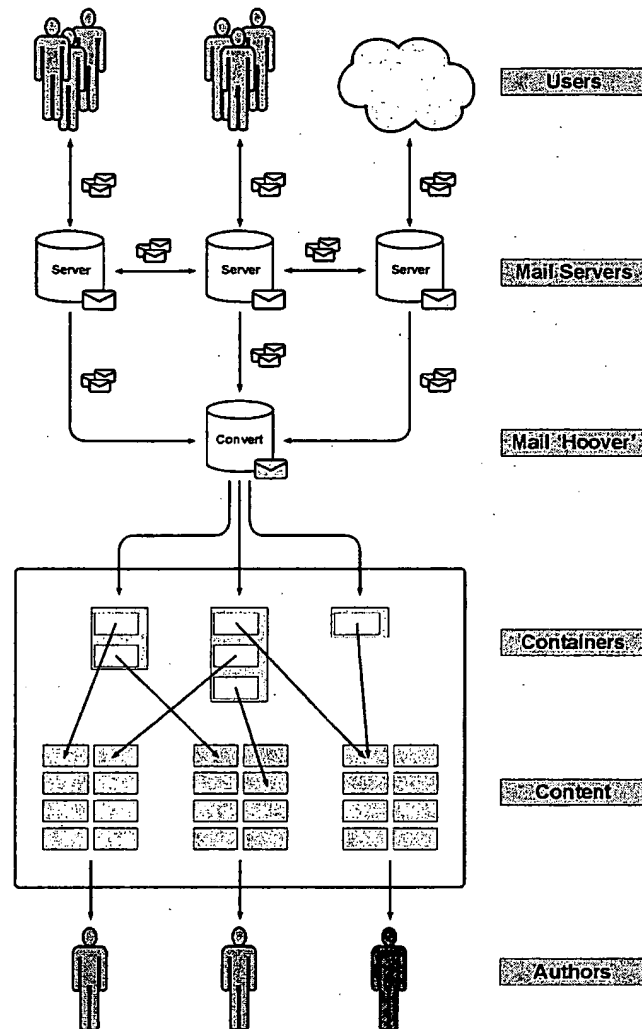


Figure 4 – Legacy migration

Figure 4 shows each email being 'worked back' to its constituent messages, and each of these messages being stored as single instance respective to their authors.

Techniques for downstream analysis are discussed in Section 6.

3.3 Summary

BlueSpace legacy migration is a significant piece of computational work, and may be well suited to a Service Integrator that can reuse the infrastructure required. Corporate migration from a legacy mail system such as Microsoft Exchange to a BlueSpace system can happen in an evolutionary manner, smoothing the computational spike associated with a rapid conversion. In addition, legacy migration is suited to grid-based analysis work, and can thus be carried out over a period of time using redundant cycles on desktop machines, as discussed in Section 6.

The cost impact of legacy migration and moving a mail system to BlueSpace is dramatic. A corporation may not need to buy new storage for their email environment for three years, as their freed up capacity may prove to be enough for all growth in that period. And this cost saving is in addition to the business value in having better access to the historical mail content, and treating it as a knowledge system.

4 Email Search

4.1 Purpose

A purpose of BlueSpace email search is to allow corporations the opportunity to gain many of the advantages of BlueSpace with respect to knowledge management and search capabilities, without needing to 'turn off' their existing email infrastructure. Corporations can still use Exchange as a mail platform, but can deploy BlueSpace email search alongside it to allow users to search the email knowledge system effectively. This has significant benefits:

- Corporations do not need to significantly affect their existing email infrastructure – BlueSpace email search can be implemented alongside existing email systems.
- Corporations can open their existing email systems to enterprise wide searching, making the knowledge captured in the email system available to the enterprise – not just each person who sent and received each email.
- By forcing more regular deletion patterns on the existing email environment, corporations can obtain storage and backup savings by relying on BlueSpace email search for long-term email management.
- The same governance model used by BlueSpace to manage the TransMail environment can be applied to email content, particularly in respect to retention of mail content by user.

4.2 Search Rights

Rights can be based on participating communicating parties in a rules engine, or on reporting structures. It can also be based on advanced criteria such as when messages are received by sets of users or on keyword analysis. In addition, search rights can be based on existing enterprise meta data, such as that for attachments or embedded in emails. For example, if files are already captured in a document management system, an enterprise might decide to allow collaborative access to emails with these file attachments in a way that mirrors collaborative access to the files in the document management system.

4.3 Downstream And Upstream

In order to deploy BlueSpace email search, the BlueSpace application attempts to 'piece back together' the context of the original communications. As shown in Figure 3, whilst legacy migration occurs purely on historical content, BlueSpace email search occurs both on historical content and on real-time content being created by users. This new content must also be added to the BlueSpace email search repository in order to keep it up to date. Given this, both downstream analysis and upstream structuring techniques are available, as described in Section 2.6. Downstream analysis in the context of email search is summarised in Section 4.4, whilst upstream structuring is discussed in Section 4.5.

4.4 Downstream Analysis

Figure 5 shows a deployed BlueSpace email search system based purely on downstream analysis. This is very similar to that shown for BlueSpace legacy

migration in Figure 4, the main difference being making the search functionality directly available to users, typically through a browser-based interface.

Search facilities can also be made available through the mail client by use of a plug-in application. The plug-in allows the same search facility in a mail client such as Outlook to access both a user's local mail archive, and also the BlueSpace search archive including collaborative search of email sent by other users.

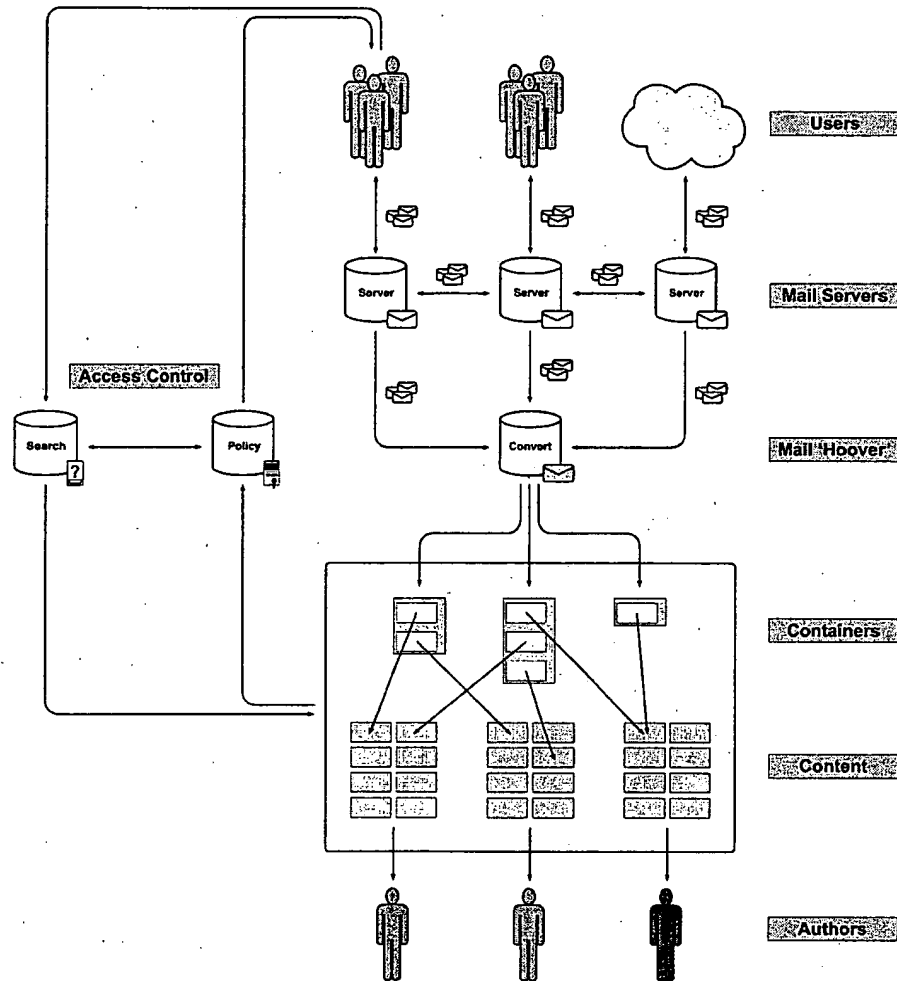


Figure 5 – BlueSpace email search

4.5 Upstream Structuring

Upstream structuring of data requires interaction of BlueSpace email search with email being sent by users. There are two levels of deployment for upstream structuring:

- A corporation might deploy upstream structuring purely for emails sent outside the BlueSpace email search environment. There are two further scenarios here:
 - In an enterprise that has deployed BlueSpace email search for all email servers, this would imply all email sent through the internet gateway to external contacts.

- In an enterprise that has deployed BlueSpace email search for a subset of email servers, this would imply all external email outside the company and also all email sent to users inside the company but outside the BlueSpace email search environment.
- A corporation might deploy upstream structuring for all emails sent by users in the BlueSpace email search environment. This would include external contacts, internal contacts outside the environment, and also internal contacts inside the BlueSpace email search environment.

Tagging external email content with embedded references in upstream structuring is particularly suited to gateway appliance deployments. With a BlueSpace email search gateway appliance, all email that passes through is tagged with the required reference numbers. This is similar to passing all email content that leaves the company through a content filter to scan for company confidential information. But rather than scanning the content passively, the BlueSpace email search gateway appliance modifies each piece of content to allow it to be tracked.

There are three topologies and associated modes of operation for upstream structuring:

1. Single gateway
2. Multiple gateways
3. Email server plug-in

4.5.1 Single gateway

The first type, single gateway, is shown in Figure 6.

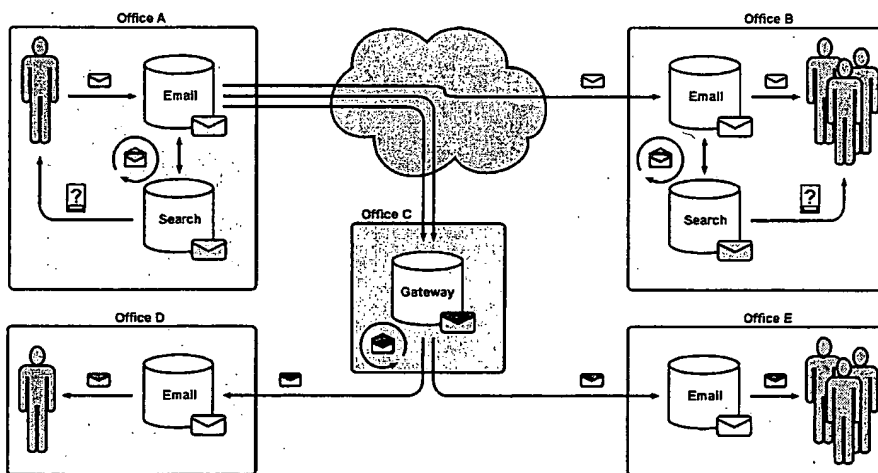


Figure 6 – Single gateway

In the scenario in Figure 6, BlueSpace email search appliances are positioned alongside a set of email servers – those in Office A and Office B. These appliances conduct downstream analysis of email content. Emails external to the BlueSpace email search environment are tagged using upstream structuring to facilitate more efficient downstream analysis. This tagging is done by a BlueSpace email search gateway appliance in Office C. Office D might be internal to the company but outside their BlueSpace email search environment, Office E might belong to another corporation.

Note that the gateway communicates with the search appliances in order to pass over the tags it has associated with each content fragment.

The system in Figure 6 is particularly suited to environments in which the BlueSpace email search deployment is large, and a gateway appliance can be positioned alongside existing internet gateway filters with relative ease.

4.5.2 Multiple gateways

The second type, multiple gateways, is shown in Figure 7.

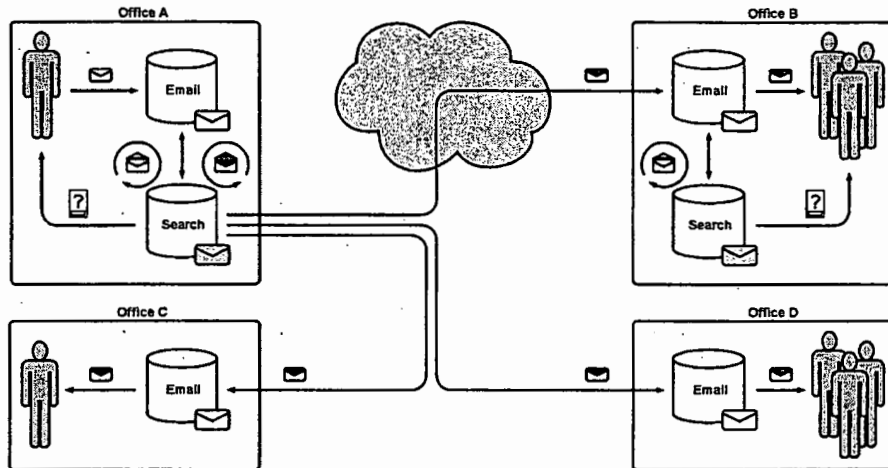


Figure 7 – Multiple gateways

In Figure 7, each search appliance conducts both downstream analysis and upstream structuring. This leads to all email having upstream structuring tags embedded into it. In any given office, a search appliance may handle mail from more than one email server. In a very centralised mail deployment, where perhaps two hundred email servers are run from a single data centre, the search appliance would become a single 'search instance'. This search instance would contain many servers acting in a large cluster, similar to the scaling approach deployed in the BlueSpace TransMail application.

4.5.3 Email server plug-in

The third type, email server plug-in, is shown in Figure 8.

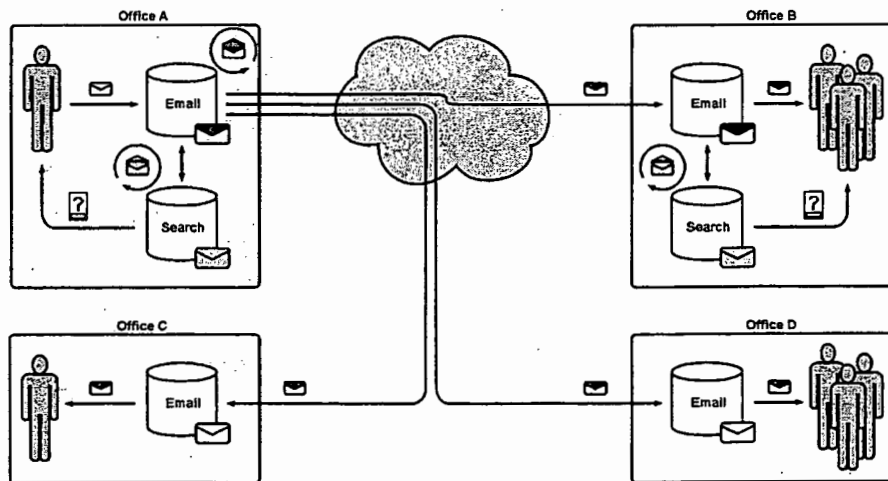


Figure 8 – Email server plug-in

In Figure 8, each email server in the BlueSpace email search environment carries out upstream structuring itself. Upstream structuring, described in

detail in Section 7, can be performed by adding another line of meta text into the email headers of each email sent. Microsoft Exchange and Outlook already exhibit custom email header fields with tags such as 'X-Priority:' to indicate the priority of an email. Whilst this information is hidden from end users by most email clients such as Microsoft Outlook, it is still available for viewing (via the mail window, View Options menu – headers are shown in the 'Internet headers:' window).

4.6 Search Criteria

Having a relational mail architecture allows rapid searching of content, and particularly improves the ability to perform queries with respect to message objects and not just emails themselves. Human readable search requests include:

- "Show all emails I received that contain this message fragment."
- "Show all emails I sent that contain this message fragment."
- "Show every email sent or received by the Investment Bankers set of users that contains this message fragment."
- "Show every recipient who received this message fragment."
- "Show the entire path of this message fragment through the email system."
- "How many unique emails contained this message fragment?"

4.7 Search Relevance

In addition to allowing a user to perform more contextual searches, performing relational analysis on email content helps discover a new piece of information – 'popularity' – that can improve search relevance. The email search appliance 'knows' how many times each message fragment has been included in different emails. It also 'knows' the recipients of those emails, and their subsequent actions.

5 SMTP Gateway

5.1 Introduction

BlueSpace TransMail is a powerful internal mail and messaging tool, allowing full auditing of content and substantial infrastructure cost reduction. However, people need to communicate with external contacts, and a BlueSpace TransMail server should have a gateway to legacy systems via Simple Mail Transport Protocol (SMTP).

Three methods for external mail were introduced above:

1. Sending a notification email and requiring the recipient to click a link to view the TransMail from the senders mail server as a web page. This method is suited to small volumes of content where content control is of paramount importance. An example would be an accounting firm consulting with a company's board as part of an audit – either the accounting firm or client company could host the other parties on their own BlueSpace system.
2. Wrapping up a TransMail in an SMTP envelope and sending it as a normal email to the recipient. For corporations that purchase BlueSpace due its dramatic impact on the TCO of email, requiring all external parties to be hosted as guests and have to click links each time to view content would not be attractive.
3. Discovering that the receiving mail server is also a BlueSpace server, and then switching to a BlueSpace protocol based on web services. This clearly occurs only between BlueSpace servers.

This section focuses on the BlueSpace SMTP gateway.

5.2 Why Bother?

A question to ask in the context of the BlueSpace SMTP gateway, is what is the point of attempting to structure external (incoming) mail content. Each BlueSpace TransMail could be simply dumped to an SMTP email, and then an incoming reply or forward could be treated as a new object in its entirety. This would then mean that the system would not need to undertake the hassle of attempting to work out whether some of any incoming email already existed in the BlueSpace content store.

There are several problems, however, with just creating all incoming emails as new message objects in their entirety:

- **Duplication:** Some content from incoming emails would already exist in the content store, thus causing unnecessary duplication in infrastructure.
- **Governance:** As soon as content objects in the database become duplicated without the relational links being applied, guaranteeing access control and retention over those content objects becomes impossible. For example, a submessage (BlueSpace terminology for a message written by a user as illustrated in Figure 1) written by an analyst may not be able to be read by an investment banker internally. But if a user forwards the submessage as part of an SMTP email externally, and then the external recipient forwards it to an analyst, the Chinese Wall fails because the system would treat the whole email from the external party as a new content object.

- **Analytics:** The lack of relational structuring of mail content at the external interface would degrade the information available to BlueSpace to conduct analytics on external mail interactions. Such analytics can be very valuable, as it deals with understanding a company's customer and partner interactions. This also affects TransMail search functionality, as well as a user's ability to see the lifecycle of a submessage they have written.
- **Trust:** One of the key features of the BlueSpace relational mail environment is the trust that it introduces into mail history – every submessage in the history is stamped and audited, guaranteeing authenticity to the end user. Clearly at the external interface, the trust is lessened as email has no security or trust model. But if no relational restructuring is attempted on content written by the parties in the email history that come from inside the organisation, an employee has no ability to know whether this content is authentic or has been modified.
- **Workflow:** In BlueSpace Communication Workflow Product (CWP) deployments, accepting the way such workflow information has been interpreted/translated by third party 'text only' gateways may involve losing much of that information. This may cause problems in the process chain.

For these reasons, BlueSpace re-establishes as much of the relational structure as is possible for incoming email to BlueSpace TransMail servers.

5.3 Nature Of Interaction

As already highlighted in Figure 3, the BlueSpace SMTP gateway does not handle any historical content (this is handled by the BlueSpace legacy migration application), but rather handles real-time mail content. The real-time nature of the SMTP gateway interaction determines that both downstream analysis and upstream structuring are available (and deployed) as techniques to enforce relational structuring on to external email content.

5.4 Content Flow

A scenario involving the BlueSpace SMTP gateway is shown in Figure 9.

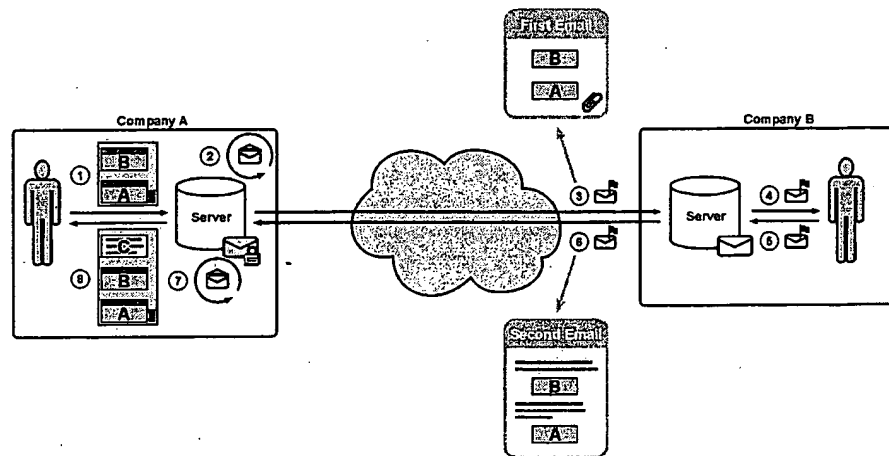


Figure 9 – BlueSpace SMTP gateway

The steps in Figure 9 are as follows:

1. A user in company A sends a TransMail to an external recipient.

2. The BlueSpace TransMail server in company A wraps the TransMail up in an SMTP envelope. This step employs upstream structuring.
3. The BlueSpace TransMail server in company A sends an email containing the TransMail content to the mail server in company B.
4. The mail server in company B delivers the email to the recipient.
5. The recipient in company B replies to the email.
6. The mail server in company B sends the replying email to the BlueSpace TransMail server in company A.
7. The BlueSpace TransMail server in company A unwraps the SMTP envelope, rediscovering the two submessages in the original TransMail, and placing the new content from the external user into a new content object – submessage C. This step employs downstream analysis.
8. The BlueSpace TransMail server in company A delivers the TransMail to the user in company A.

Note that when external content is received, the BlueSpace user interface distinguishes internal TransMail content (that has guaranteed authenticity) and external email content (with weak authenticity). This is represented in Figure 9 by submessages A and B being coloured blue, whilst submessage C is coloured yellow to indicate its email origin.

The scenario in Figure 9 is relatively easy to manage, as the email that was analysed was a direct reply to the original TransMail. Examples in which a TransMail is sent externally, forwarded within another company, and then forwarded to a different employee in the originating company are somewhat more difficult to analyse. Technical techniques are discussed in Sections 6 and 7, but the principles of author analysis and content representation are first covered in Section 5.5.

5.5 Author Analysis

Two core questions in author analysis are:

1. To what level is author analysis attempted, including the minimum size of fragment, and to what degree external author analysis is attempted?
2. In cases in which author analysis may have ended up with a result that appears to be errant to a recipient, how can a recipient view the way the email actually appeared when received?

In Section 5.4, analysis was conducted at the submessage level, and all new content was distilled out of the replying email in step 7 and placed into a separate submessage object – C. If, however, an external recipient had made many comments throughout the original submessages (A and B in Figure 9), removing the new content and presenting it separately may make it very difficult to understand the context of what is being said. To cater for such scenarios, the user interface provides a way to toggle between viewing the email as it was received and viewing the email as a TransMail with each content object separated.

A further question presents itself when an external user, such as the recipient in company B in Figure 9, forwards only part of a submessage to a user inside the company A in Figure 9. If the recipient in company A has already seen the submessage that has been forwarded (as would be the case as depicted in Figure 9), then there is no reason not to show all history submessages in full. However, if the recipient of the external email in Figure 9 were a different user, should this other user in company A be shown historical content that they have not been privy to before? This can be determined in BlueSpace as an

enterprise-wide preference by a Compliance Officer (a super-user role within BlueSpace that has associated administrative permissible actions). Either the full submessages are shown, or only the parts of the submessages that the user has received directly are shown.

Note that if a user is not allowed to view the incoming submessage (e.g. the investment banker and analyst scenario mentioned in Section 5.2), the system will prevent them from seeing the submessage. The behaviour in this scenario is again determined by the administrative preferences. Either the email without the content that would contravene the policy is shown, or no content is shown and the email is placed in a queue awaiting viewing by a designated officer of the company, e.g. a Compliance Officer. Note that this mediating behaviour is already implemented in the email community, such as when a user receives an encrypted email and it is filtered by a virus checker and placed in a queue awaiting action from an administrator.

The level of granularity to which content recognition is attempted is an administrative preference, and different corporations do require different levels. For example, a corporation that is using BlueSpace for a highly regulated environment such as an investment bank may require much greater granularity than a corporation that is using BlueSpace to achieve cost savings. A bank may take on a \$100m liability if a paragraph of text written by an analyst is forwarded to an investment banker. By way of comparison, a retail company using BlueSpace to reduce storage requirements would likely be content with 80% external content recognition.

5.6 Summary

The BlueSpace SMTP gateway provides a mechanism for BlueSpace TransMail users to communicate with external parties, without causing significant content duplication and enterprise risk. Content analysis, involving rediscovering the contextual threads in inbound email, also improves the user experience (by providing a better picture of what's going on).

6 Downstream Analysis

6.1 Introduction

Downstream analysis is the process that is carried out on email content some time after it has been written. The downstream analysis process inspects all email in an email repository (such as an Exchange environment), and then places the content into a BlueSpace store where it can be used for different purposes, including querying. Note that downstream analysis is also employed when handling incoming emails through an SMTP gateway.

This section discusses the technical implementation of such a downstream analysis process. As different email clients behave differently, there is a discussion on how emails are formatted and constructed. Next, the TransMail output of the process is described followed by an in depth look at the downstream process itself.

6.2 Email Construction, Storage and Formatting

Emails are text strings that follow particular conventions. When these conventions are followed, email servers are able to process and relay these emails so that they eventually arrive at their destination. This section looks at how emails are constructed, formatted and stored.

6.2.1 Construction

Emails are strings of 7-bit characters. Immediately this causes a problem when emails are required to send 8-bit binary information such as images and documents as they need to be converted to 7-bit and appended to the email body. These multipart emails are still text strings, so email clients need to know how to parse them. The Multipurpose Internet Mail Extensions (MIME) standard is used to determine the structure of email and allows attachments to be seen as separate to the mail text.

Along with the body of the email are headers. These headers contain information such as:

- Sender name and email address
- Recipient name and email address
- Date sent
- Unique Message-Id
- Email client of the sender
- Routing information
- References and 'In Reply To' identifiers

Note that the headers available are largely determined by individual email servers and email clients that use them. Headers cannot always be guaranteed to be present.

When email is sent, the client simply contacts the server and tells the server who the sender is, where the mail should go, and then transfers the text string that is the email.

6.2.2 Email Body Structure

The structure of the body of an email plays a significant role in downstream analysis. The behaviour of email clients determines the degree to which submessages can be identified within the body of an email.

On a reply or forward event, some mail clients will not append any of the original mail to the reply or forwarded message. Some mail clients will append the original message. They use delimiters followed by the mail, for example:

```
-----Original Message-----
lines from the original message
```

In some configurations the message will also include a subset of header information:

```
-----Original Message-----
From: John Doe [mailto:john.doe@mycompany.null]
Sent: 02 June 2004
To: Jane Doe
Subject: A message

Lines from the original message
```

The exact formatting will change from mail client to mail client, and also with the individual settings set by users. An example of a similar strategy from a different mail client is given below:

```
"John Doe" <john.doe@mycompany.null> on 01/06/2004 13:14:18

To: "Jane Doe" <jane.doe@mycompany.null>
cc:

Subject: FW: message

Lines from the original message
```

Some clients prefix each line with a indentation character and start the quotation with a statement of authorship, for example:

```
On 26th May 2004 at 15:10, John Doe wrote:

> Lines from the original mail
```

This is problematic as the mail client may also wrap characters at a certain line length, e.g. 76 characters, resulting in an inconsistent indentation for a submessage when there are multiple replies, for example:

```
> > hostel is happy for you to take only 5 beds out of the six.
> However I was
> > told that this dorm needs to be booked soon, as they tend to go
> quickly -
> > the guy I spoke to said that we probably have a week before
```

A procedure for being able to determine the original content is required, and is discussed below in Section 6.3.3.1.

6.3 Conversion

6.3.1 Stores

6.3.1.1 Data Stores

There are four data stores used in the basic conversion process:

1. User store
2. Email store
3. TransMail store
4. Submessage store

6.3.1.2 User Store

The user store simply keeps track of user names and email identities.

6.3.1.3 Email Store

The email store provides the capability of iterating through emails and retrieving the following information regarding each email:

- 'From' email address
- 'From' name
- 'To' recipients email addresses
- 'To' recipients names
- 'Cc' recipients email addresses
- 'Cc' recipients names
- Body
- Subject
- Date sent
- Date received
- Headers
- Attachments
- User and folder in which the email was located.

As stated earlier, not every mail store will be able to provide all information. For details regarding a minimum set, see earlier sections.

6.3.1.4 TransMail Store

The TransMail store allows TransMails to be added to it. In implementation terms, this would be a resizable array such as a Vector.

6.3.1.5 Submessage Store

The submessage store provides a mechanism for associating unique and 'original' content with submessages. A submessage consists of:

- Reference to a sender identity
- References to 'To' identities
- References to 'Cc' identities

- 'From' email address and name*
- 'To' email addresses and names*
- 'Cc' email addresses and names*
- Body
- Subject
- Date sent
- Date received
- References to attachments
- 'Original' flag*
- Body hash*

The fields marked with a "*" are fields that are not normally associated with 'normal' submessages, but are required for this conversion process. Once the conversion process is complete, these fields can be ignored.

The submessage store is simply a mapping of the body hash field to the submessage itself. This store allows a submessage to be added, and for existing submessages to be retrieved by supplying a body hash as a key. The reason for this design is described later.

6.3.2 Procedure

Pseudo-code is used to describe the procedure of converting emails to TransMails.

```

For each email, e, in the email store {
    Determine if e contains history
    If e does not contain history
        ProcessNonComposite(e)
    If e contains history
        ProcessComposite(e)
}

```

There are several ways in which email can be processed in order to determine if it contains history. These are described in section 6.3.3.3.

The above procedures 'ProcessNonComposite' and 'ProcessComposite' are described in sections 6.3.2.1 and 6.3.2.2 respectively.

6.3.2.1 Processing a Non-Composite Email

A new TransMail envelope is created. Next, a new submessage is created, copying available values from the email to the submessage, including:

- 'From' email address and name
- 'To' email addresses and names
- 'Cc' email addresses and names
- Body
- Subject
- Date sent
- Date received

Note that user identities may not have been determined for the sender and all recipients, so for now the process just simply copies over the email addresses and names to the submessage.

The submessage is flagged as 'original', meaning that this message is not the result of a reply or forward. A message body hash is also computed, discussed in Section 6.3.3.1 below, which is linked to the submessage object.

Finally, attachments are decoded using a MIME filter, and added to the attachments store (see Section 6.3.3.2). References to these attachments are linked to the submessage.

6.3.2.2 *Process a Composite Email*

Processing a composite email is a matter of determining the individual components of the email. Section 6.2.2 describes how various email clients create new messages based on old messages.

A TransMail container is created that will reference the submessages.

The first submessage will be 'original', but subsequent parts will not. For each of the history submessages, an attempt will be made to find the original. This is done by taking each message body hash of each part in turn (see 6.3.3.1) and determining if the hash already exists. If it does, then the TransMail container can point to the submessage. If the submessage does not already exist, then a new submessage is created and marked as 'not original'.

A post-process can then iterate through non-original submessages and resolve them if their originals are found.

6.3.3 Procedures and Functions

6.3.3.1 *Message Body Hash*

The message body hash reduces a message body into a representation of it that does not contain any extra client formatting such as chevron indentations. A mechanism used to achieve this is to take all alphanumerical characters, and ignoring all others.

6.3.3.2 *Attachments Store*

The attachments store has the property that each unique attachment is only stored once. This is achieved by associating each attachment with a hash value. This hash value can be generated by a cryptographic function such as the MD5 or SHA-1 algorithms.

When a new attachment is to be added to the store, the store first calculates the hash for that attachment and determines whether that hash is already exists. If so, the attachment need not be restored. If the hash is not found, then the attachment is saved, as well as the hash value associated with it.

6.3.3.3 *Determining Composite and Non-Composite Emails*

A measure to determine if an email is composite or non-composite is to have a register of known email client behaviour, and searching for text patterns. For example, if the subject starts with 'Re:' or if the message body contains '--- Original Message ---'. A more advanced approach associates matches with a score, and if the resulting score is above a threshold, then the email can be said to be composite.

7 Upstream Structuring

7.1 Tagging

In order for BlueSpace to recognise inbound emails as originating from a BlueSpace environment, outbound emails must be tagged in such a way that email clients include those tags when a reply or forward action is taken.

Tagging of outbound email depends on the type of email, either plaintext or HTML email. HTML emails are more suited to tagging as they allow meta-tags to be included in the email body without the user's email being interrupted.

7.1.1 Plaintext Tagging

If emails are to be sent as plaintext, tags are used to identify each individual message in the email. Also, a header is added to the email along with the standard email headers. Examples of how to tag outbound emails are given below.

7.1.1.1 Email with single message

In this example, Jane Doe is using BlueSpace. She sends a message to John Doe who is not using BlueSpace, but is an email user. Jane's BlueSpace server determines that the message should be sent as plaintext email.

```
Mon, 16 Sep 2004 00:01:02 +0300
To: "John Doe" <john.doe@otherhost.net>
From: "Jane Doe" <jane.doe@bluespace.host.net>
Subject: Lunchtime meeting
X-Mailer: BlueSpace SMTP Gateway v3.01.7788
X-Priority: 3 (Normal)
X-BLSP-V: 3.01.778
X-BLSP-ID: <SRV1MMBuado7fusvV6v00000003@blspgateway1.host.net>
Return-Path: jane.doe@bluespace.host.net
Message-ID: <SRV1MMBuado7fusvV6v00000003@blspgateway1.host.net>
Date: 16 Sep 2004 00:01:02 +0300
```

Dear John,

Hope all is well with you. Lets meet this lunchtime.

Regards,

Jane.

```
-----BEGIN BlueSpace ID-----
Version: 3.01.778
BLSP-ID: SRV1MMBuado7fusvV6v00000003@blspgateway1.host.net
SUBID: iQCVaWUBMJrRF2N9oWBghPDJAQE9UQQAt1@blspgateway1.host.net
-----END BlueSpace ID-----
```

The headers section of the email includes an X-BLSP-ID. Also, the email body has been tagged by the BlueSpace gateway. The tag in the body identifies the following pieces of information:

- The version of the BlueSpace system that sent the message
- The BlueSpace identifier for the TransMail
- The BlueSpace identifier for the Submessage.

The mechanism by which the tag is formatted is not important – it is the information that it represents that is particularly important.

7.1.1.2 Email with single submessage and attachments

In this example, Jane sends John two documents from her BlueSpace account.

```
Mon, 16 Sep 2004 00:01:02 +0300
To: "John Doe" <john.doe@otherhost.net>
From: "Jane Doe" <jane.doe@bluespace.host.net>
Subject: Meeting pre-reads
X-Mailer: BlueSpace SMTP Gateway v3.01.7788
X-Priority: 3 (Normal)
X-BLSP-V: 3.01.778
X-BLSP-ID: <SRVlMMBuado7fusvV6v000000003@blspgateway1.host.net>
X-BLSP-AT: qLQRkYNoBActFBZmh;PIcEmI5iFd9boEgvp
Return-Path: jane.doe@bluespace.host.net
Message-ID: <SRVlMMBuado7fusvV6v000000003@blspgateway1.host.net>
Date: 16 Sep 2004 00:01:02 +0300
```

Dear John,

Please find the two documents attached.

Regards,

Jane.

```
-----BEGIN BlueSpace ID-----
Version: 3.01.778
BLSP-ID: SRVlMMBuado7fusvV6v000000003@blspgateway1.host.net
SUBID: iQCVAwUBMJrRF2N9oWBghPDJAQE9UQQAt1@blspgateway1.host.net
ATTACHMENTS: qLQRkYNoBActFBZmh;PIcEmI5iFd9boEgvp
-----END BlueSpace ID-----
```

A line is included in the headers to represent the two attachments in the email and the BlueSpace ID tag also lists the attachment identifiers.

7.1.1.3 Email with multiple messages

In this example, Jane forwards a message that contains history to John. The email is required to insert identifiers for each submessage.

```
Mon, 16 Sep 2004 00:01:02 +0300
To: "John Doe" <john.doe@otherhost.net>
From: "Jane Doe" <jane.doe@bluespace.host.net>
Subject: FWD: Re: Agenda
X-Mailer: BlueSpace SMTP Gateway v3.01.778
X-Priority: 3 (Normal)
X-BLSP-V: 3.01.778
X-BLSP-ID: <SRV1MMBuado7fusvV6v00000003@blspgateway1.host.net>
Return-Path: jane.doe@bluespace.host.net
Message-ID: <SRV1MMBuado7fusvV6v00000003@blspgateway1.host.net>
Date: 16 Sep 2004 00:01:02 +0300
```

Dear John,

Would you be able to attend this meeting?

Regards,

Jane.

```
-----BEGIN BlueSpace ID-----
SUBID: iQCVawUBMJrRF2N9oWBghPDJAE9UQQAt1@blspgateway1.host.net
-----END BlueSpace ID-----
```

```
-----Original Message-----
From: Michael Smith
Sent: 15 Sep 2004 13:45
To: Jane Doe
Subject: Re: Agenda
```

Jane,

Could you make sure your team has been invited.

Thanks,

Michael

```
-----BEGIN BlueSpace ID-----
SUBID: boEgvpHtIREEqLQRkYNbAIREEqFBZm@blspgateway1.host.net
-----END BlueSpace ID-----
```

```
-----Original Message-----
From: Jane Doe
Sent: 14 Sep 2004 16:23
To: Michael Smith
Subject: Agenda
```

Michael,

I've been advised of the meeting tomorrow - my team will be able to attend if required.

Regards,

Jane.

```
-----BEGIN BlueSpace ID-----
Version: 3.01.778
BLSP-ID: SRV1MMBuado7fusvV6v00000003@blspgateway1.host.net
SUBID: AK8AFHA29ZMCAa29aF09aRtPLm900aSVnj@blspgateway1.host.net
-----END BlueSpace ID-----
```

It is not necessary to duplicate the Version and BLSP-IDs in all submessages, so in this example only the last submessage contains this extra information.

7.1.2 HTML Tagging

HTML-based email offers an environment in which a much richer set of tagging can be used without impacting the readability of the message itself. HTML-based tagging utilises a browser's ability to not display unrecognised tags, and to include these tags when replying or forwarding.

A simple HTML email message might look like this:

```
Mon, 16 Sep 2004 00:01:02 +0300
To: "John Doe" <john.doe@otherhost.net>
From: "Jane Doe" <jane.doe@bluespace.host.net>
Subject: FWD: Re: Agenda
Return-Path: jane.doe@bluespace.host.net
Message-ID: <SRVIMMBuado7fusvV6v00000003@blspgateway1.host.net>
Date: 16 Sep 2004 00:01:02 +0300

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<BODY>
<P>Hello</P>
</BODY>
</HTML>
```

BlueSpace uses tags to tag the entire message, the individual submessages, all attachments, and also individual paragraphs. The following example utilises all of these:

```
Mon, 16 Sep 2004 00:01:02 +0300
To: "John Doe" <john.doe@otherhost.net>
From: "Jane Doe" <jane.doe@bluespace.host.net>
Subject: FWD: Re: Agenda
X-Mailer: BlueSpace SMTP Gateway v3.01.7788
X-Priority: 3 (Normal)
X-BLSP-V: 3.01.778
X-BLSP-ID: <SRVIMMBuado7fusvV6v00000003@blspgateway1.host.net>
Return-Path: jane.doe@bluespace.host.net
Message-ID: <SRVIMMBuado7fusvV6v00000003@blspgateway1.host.net>
Date: 16 Sep 2004 00:01:02 +0300

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><BODY>
<!--BEGIN BLSP-ID SRVIMMBuado7fusvV6v00000003@blspgateway1.host.net-->
<!--BLSPV 3.01.778-->
<!--BEGIN BLSPSUBID iQCVawUBMJrRF2N9oWBghPQAt1@blspgateway1.host.net-->
<P><!--BLSP P1S-->Dear John,<!--BLSP P1E--></P>
<P><!--BLSP P2S-->Would you be able to attend this meeting? <!--BLSP
P2E--></P>
<P><!--BLSP P3S-->Regards,<!--BLSP P3E--></P>
<P><!--BLSP P4S-->Jane.<!--BLSP P4E--></P>
<!--END BLSPSUBID iQCVawUBMJrRF2N9oWBghPQAt1@blspgateway1.host.net-->
<!--BEGIN BLSPSUBID boEgvpHtIREEqLQRkYNoBA@blspgateway1.host.net-->
<BR><BR><BR>
<P><!--BLSP E1S-->-----Original Message-----<!--BLSP E1E--></P>
<P><!--BLSP E2S-->From: Michael Smith<!--BLSP P2E--></P>
<P><!--BLSP E3S-->Sent: 15 Sep 2004 13:45<!--BLSP E3E--></P>
<P><!--BLSP E4S-->To: Jane Doe<!--BLSP E4E--></P>
<P><!--BLSP E5S-->Subject: Re: Agenda<!--BLSP E5E--></P>
<P><!--BLSP P1S-->Jane, <!--BLSP P1E--></P>
<P><!--BLSP P2S-->Could you make sure your team has been invited. <!--
BLSP P2E--></P>
<P><!--BLSP P3S-->Thanks, <!--BLSP P3E--></P>
<P><!--BLSP P4S-->Michael<!--BLSP P4E--></P>
<!--END BLSPSUBID boEgvpHtIREEqLQRkYNoBA@blspgateway1.host.net-->
<!--BEGIN BLSPSUBID AK8AFHA29ZMCAa29aFSVnj@blspgateway1.host.net-->
<BR><BR><BR>
<P><!--BLSP E1S-->-----Original Message-----<!--BLSP E1E-->
<P><!--BLSP E2S-->From: Jane Doe<!--BLSP E2E-->
<P><!--BLSP E3S-->Sent: 14 Sep 2004 16:23<!--BLSP E3E-->
<P><!--BLSP E4S-->To: Michael Smith<!--BLSP E4E-->
<P><!--BLSP E5S-->Subject: Agenda<!--BLSP E5E-->
<P><!--BLSP E6S-->Michael, <!--BLSP E6E-->
```

```

<P><!--BLSP P1S-->I've been advised of the meeting tomorrow - my team
will be able to attend if required. <!--BLSP P1E-->
<P><!--BLSP P2S-->Regards, <!--BLSP P2E-->
<P><!--BLSP P3S-->Jane. <!--BLSP P3E-->
<!--BLSP-ATTACHMENTS qLQRkYNoBActFBZmh;PicEmISiFd9boEgvp -->
<!--END BLSPSUBID AK8AFHA29ZMCAa29aFSVnj@blspgateway1.host.net-->
<!--END BLSP-ID: SRV1MMBuado7fusvV6v000000003@blspgateway1.host.net-->
</BODY>
</HTML>

```

Table 1 shows the elements used in the construction of this mail:

BEGIN BLSP-ID	Start of a BlueSpace mail
END BLSP-ID	End of a BlueSpace mail
BEGIN BLSPSUBID	Start of a BlueSpace submessage
END BLSPSUBID	End of a BlueSpace submessage
BLSPV	BlueSpace SMTP gateway version
BLSP P1S	Start of a paragraph of content
BLSP P1E	End of a paragraph of content
BLSP E1S	Start of an extra paragraph (not direct content, but extra formatting.)
BLSP E1E	End of an extra paragraph
BLSP-ATTACHMENTS	List of attachments associated with submessage.

Table 1 – BlueSpace structuring HTML tags

All of these elements are contained within HTML comment tags. By using this type of tag, the BlueSpace data is not displayed by the email client, but are included when the email is replied to or forwarded.

7.2 Recognition and Consolidation

Inbound email into BlueSpace is checked for upstream structuring tags. If found, the mail can then be consolidated into the BlueSpace message store, either for direct use by BlueSpace, or by the BlueSpace search facility.

7.2.1 Recognition Process

The recognition process simply looks for known BlueSpace tags in both plaintext and HTML email. If tags are found, the mail is passed over to the consolidation process.

7.2.2 Consolidation Process

Consolidation aims to add inbound messages into a TransMail store. This store can either be for a TransMail messaging platform, or for an email search facility. The TransMail store allows for the efficient storage and management of messages, and allows for business rules to apply to both searching and accessing content.

The most straightforward cases of consolidating messages is where the inbound message is an ordinary reply or forward, and no existing content has been edited, as shown in Figure 10.

7.2.2.1 Unedited Mails

Tags used in the construction of the email are used to determine whether content already exists in the TransMail database. Firstly, the TransMail id is used to determine what submessages should be present in the mail. Submessage IDs are then used to compare against the content to ensure that the content received matches the content already stored in the database.

If tags are received that indicate the content has originated from other BlueSpace servers, the consolidation process can request contact with the remote BlueSpace server to validate the authenticity of the message, and optionally retrieve the original submessage. BlueSpace servers may be configured to either allow this type of behaviour, or ignore these requests.

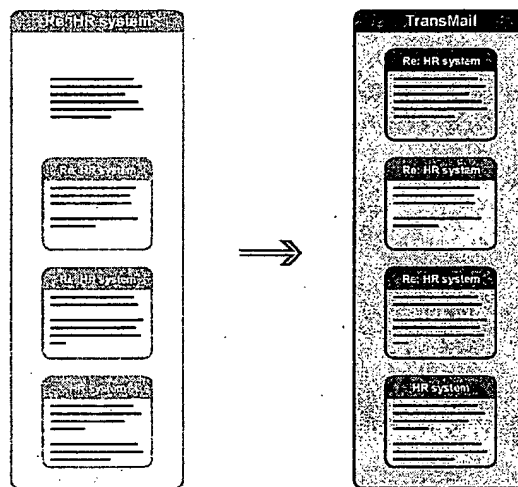


Figure 10 – Reply with separate new content

The mail should be added to the TransMail store, so a new TransMail is created. This TransMail will act as a container to the submessages that comprise the entire message.

The new email content is stored in a separate submessage repository. The submessage is then set with the subject and body of the email. The sender and recipients names and email addresses are also stored against the submessage. If the email contains attachments, these are placed into the file store. This file store only keeps one copy of the file attachments. The submessage then points to the attachments in the repository rather than the attachments being stored with the submessage.

Once the 'current' submessage has been constructed, the TransMail container can point to it. The history submessages already exist in the TransMail submessage repository, so references to them are added to the TransMail container.

7.2.2.2 Edited Mails

Sometimes authors of emails will intersperse comments throughout the mail, as shown in Figure 11. Authors may also alter the text of email, either out of necessity, or maliciously.

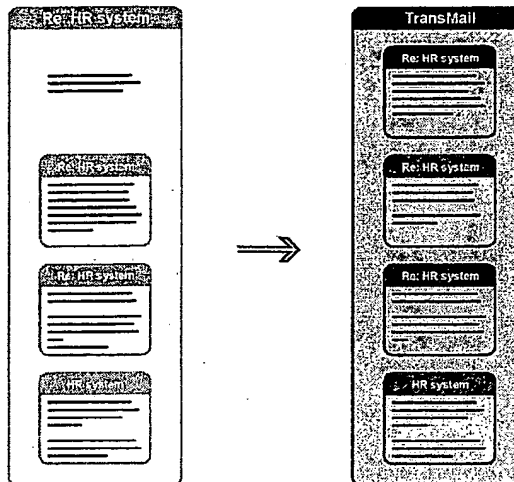


Figure 11 – Reply with interspersed new content

During consolidation, the text of a submessage may not match the original submessage stored in the database. In this scenario, the consolidation process stores the new version of the submessage as well as keeping the current version.

When the TransMail that contains the edited submessage is shown to the user, they are presented with the ability to switch between the original version of the submessage, and the received version of the submessage.

In order to achieve this, the TransMail that represents this message can refer to both original submessage content, and new submessage content. During presentation the reader is made aware that there are differences between the original and the new version.

23682/01000/SF/5128738.1

bluespace

BlueSpace Discovery Service

Contents

1.1 Index

1.1	INDEX.....	2
1.2	LIST OF FIGURES.....	2
1.3	LIST OF TABLES.....	3
2	ABSTRACT.....	4
3	E-MAIL – EARLY YEARS.....	5
3.1	SMTP AND MAIL RELAYING.....	5
3.2	MX ROUTING.....	6
3.3	SMTP VOCABULARY.....	7
3.4	RETRIEVING AND STORING E-MAIL.....	8
4	BLUESPACE OVERVIEW.....	9
4.1	INTRODUCTION.....	9
4.2	BLUESPACE ARCHITECTURE.....	9
4.3	USER ADDRESSING.....	11
4.4	CONTENT TYPES.....	12
4.5	BLUESPACE T-MAIL CACHE.....	12
4.6	IMPACT ON INFRASTRUCTURE.....	13
4.7	PROTOCOL-BASED BANDWIDTH SAVINGS.....	14
4.8	SUMMARY.....	15
5	BLUESPACE DISCOVERY SERVICE.....	16
5.1	INTRODUCTION.....	16
5.2	T-MAIL SUPERIORITY.....	16
5.3	BMTP VERSUS SMTP.....	17
5.4	BLUESPACE DISCOVERY SERVICE.....	18
5.4.1	Pre-discovery.....	18
5.4.2	Post-discovery.....	19
5.5	SERVER TOPOLOGIES.....	20
5.6	SMTP SYNTAX.....	20
5.7	BLUESPACE PRE-DISCOVERY.....	22
5.8	BLUESPACE POST-DISCOVERY.....	23
5.9	PREVENTING DISCOVERY REPETITION.....	24
5.10	E-MAIL ARCHITECTURE PROBLEMS.....	25
5.11	BLUESPACE DIRECTORY INTERFACE.....	26
5.12	BLUESPACE MAIL GATEWAY APPLIANCE.....	27
5.13	INTERNAL USE.....	30
5.14	SUMMARY.....	30

1.2 List of Figures

Figure 1 – SMTP delivery backbone.....	5
Figure 2 – Message flow through MTAs.....	6

Figure 3 – MTA routing based on MX values	7
Figure 4 – T-Mail relational structure	9
Figure 5 – Example of T-Mail flow	10
Figure 6 – MX lookup by BlueSpace to route e-mail	11
Figure 7 – BlueSpace T-Mail cache in operation	13
Figure 8 – Mail sent as both T-Mail and e-mail	16
Figure 9 – The process of pre-discovery	18
Figure 10 – The process of post-discovery	19
Figure 11 – Three topologies for external mail	20
Figure 12 – SMTP gateway architecture	25
Figure 13 – BMG handling e-mail	28
Figure 14 – BMG handling T-Mail	29

1.3 List of Tables

Table 1 – Mail network traffic comparison	15
Table 2 – E-mail versus T-Mail for external mail	17
Table 3 – Simple SMTP server to server lines	21
Table 4 – Extended SMTP server to server lines	22
Table 5 – BlueSpace pre-discovery via SMTP lines	22

2 Abstract

E-mail was invented in the early 1970s as part of an academic experiment. Whilst its fundamental protocols (such as Simple Mail Transfer Protocol – SMTP) have changed little since their inception in 1982, real-world usage of e-mail has changed dramatically. E-mail 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 e-mail is growing by 50% every year, causing companies' bandwidth requirements to double every six months, and has commented that, "The cost of supporting email, technology infrastructure, IT and so on has got to a point where CIOs are under pressure to sort it out."

SMTP treats each e-mail as a string of text similar to a postcard or telegram that must be passed between computers until it reaches its desired recipients. Mainly due to historical reasons, SMTP has achieved a near complete monopoly on sending e-mail outside a company's boundary, despite its bandwidth inefficiency, Spam vulnerability and lack of authentication or security.

BlueSpace Mail Transfer Protocol (BMTP) is far more advanced than SMTP, as it is based on a web-approach to mail transfer in which there are two forms of data – content data that is 'published' and does not change, and audit data that changes each time a user carries out an action. BlueSpace Discovery Service (BDS) allows independent BlueSpace users to send T-Mails to each other using BMTP rather than defaulting back to SMTP. There are two modes:

1. *Pre-discovery* – The sending server attempts a BlueSpace handshake with each receiving server before sending an e-mail.
2. *Post-discovery* – The receiving server discovers a BlueSpace identifier in the first e-mail from the sending server, and initiates a handshake.

Pre-discovery is typically attempted first, and if this fails, post-discovery can be attempted. In both cases, the two servers 'remember' that the other is a BlueSpace server, and in future use BMTP to communicate between them.

BlueSpace recognises that many corporate mail infrastructures use a gateway SMTP server to route e-mail to internal mail servers, which may include both conventional e-mail servers and BlueSpace T-Mail servers. The BlueSpace Mail Gateway Appliance has been specifically designed for such scenarios, intercepting incoming SMTP sessions, and then routing the SMTP session depending on the protocols supported by the external and internal mail servers.

This white paper provides a revolutionary answer to some of the questions posed today by business users of e-mail. Suzanne Sluizer, co-author of Mail Transfer Protocol, recently remarked, "I would suggest they just write a new protocol from the beginning." BlueSpace T-Mail takes up that challenge, and with BDS to effect propagation, BlueSpace has the strategy and tool with which to drive enterprise adoption.

3 E-mail – Early Years

3.1 SMTP and Mail Relaying

SMTP, authored in 1982, was based on the concept of passing messages between servers to reach their destination.

ARPANET was based on a collection of computers that connected to each other in a symbiotic manner. There was no hierarchy, and very little control. The internet today is still ultimately a collection of carrier networks that all peer with each other.

SMTP relies on this network and the trustworthiness of intermediary computers to pass a mail on until it reaches its destination – the 'store and forward' principle. Mail relaying was seen as a real advantage when SMTP was designed, as it effectively stored mail on the network for servers that had become disconnected.

SMTP delivery backbones are built around Message Transfer Agents (MTAs). In Figure 1, MTAs are the servers that forward emails to their eventual destination. A user accesses the mail system through a user agent – this could be a terminal application such as SNDMSG.

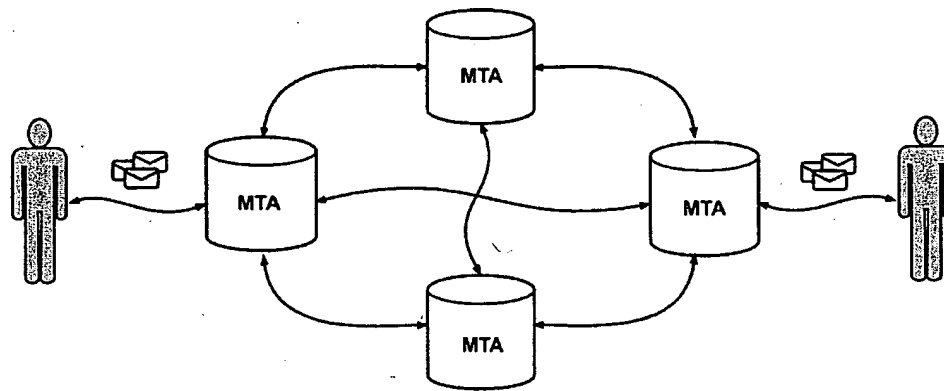


Figure 1 – SMTP delivery backbone

Whenever an MTA receives a mail, it must decide what to do with it. Is the recipient local, or does the MTA need to pass it on to another MTA, and if so, which one? The well known '@' sign is used to determine whether the user is local or not. For example, john.smith@store.com would not be local to bob.johnson57821@hotmail.com, thus requiring Hotmail to do an MX lookup to find out where to send an e-mail that Bob has written to John, as described in Section 3.2.

The diagram in Figure 2 shows message flow through MTAs.

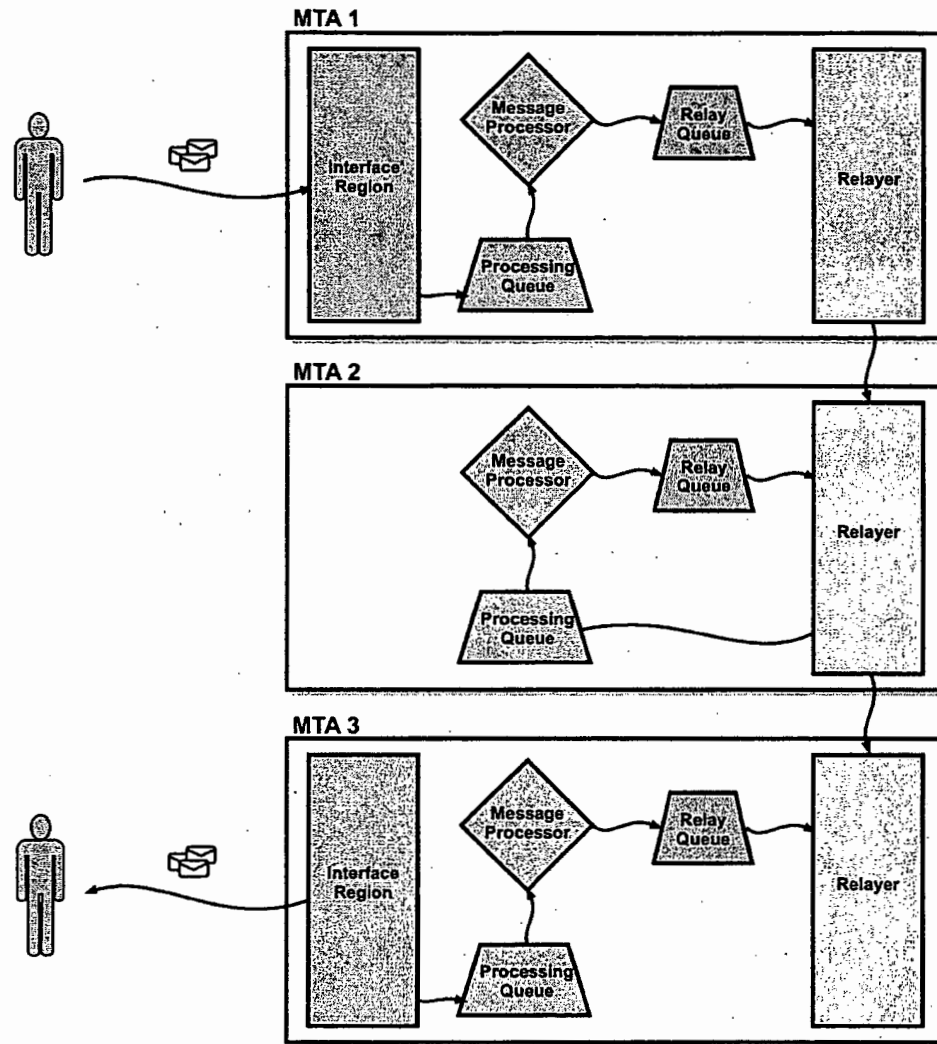


Figure 2 – Message flow through MTAs

First a user creates a mail using their user agent (e.g. a client such as Microsoft Outlook), and it is then passed to a queue for processing. When passed to the message processor, the MTA decides whether to keep the mail (for a local recipient) or to pass it to another MTA (for a remote recipient). In the diagram, the recipient is not on the same MTA as the sender, and so MTA1 relays the mail to MTA2, where it is deposited into the queue of mails waiting to be processed. On processing, MTA2 discovers that the recipient is not local, and so again it looks to relay it to another MTA – MTA3. MTA3 inspects the mail and finds that the recipient is local, so instead of relaying the mail, it places it into a delivery queue from which it is eventually given to the recipient.

3.2 MX Routing

MX records were designed to facilitate efficient e-mail routing, and are part of the Domain Naming System (DNS), which is how human readable domain names such as www.amazon.com are associated with the Internet Protocol (IP) address space that computers use (Amazon's primary IP address is 207.171.178.132). When delivering mail inside a company, a sending MTA will first try to connect

directly to the receiving MTA and then transfer the mail using an SMTP connection. If the receiving MTA (i.e. the MTA where the recipient's mail box lives) is not available, the sending MTA uses the MX record for the receiving MTA to determine which MTA to forward the mail to (i.e. which is closest to the receiving MTA). Figure 3 illustrates this in detail.

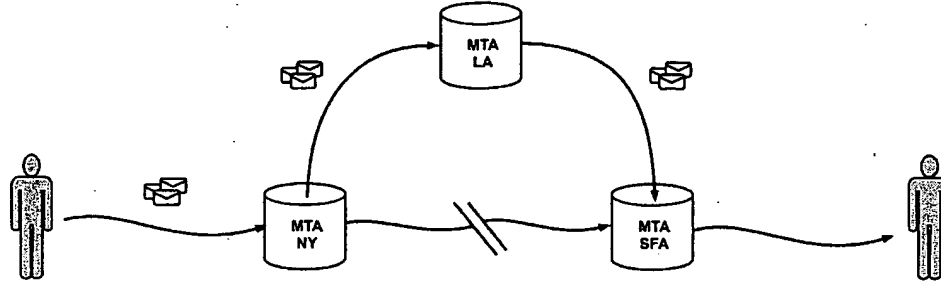


Figure 3 – MTA routing based on MX values

In this scenario, a user who works at Store & Co. in New York is trying to send an e-mail to another user who works in the San Francisco office. The MTA in New York (ny.store.com) looks up the MX record for the San Francisco MTA, and finds that the entry with the lowest MX value (10) is sfa.store.com. The New York MTA attempts to connect to sfa.store.com but it cannot reach it, as the San Francisco office have temporarily lost their internet connection.

The New York MTA then goes back to the MX record for the San Francisco MTA, and finds that the next entry with an MX value of 20 is la.store.com. The New York MTA connects to la.store.com and relays the e-mail for Sally to the Los Angeles MTA. Later that day the internet connection at the San Francisco office is restored, and the Los Angeles MTA then delivers the e-mail to sfa.store.com from which the recipient later reads it.

MX routing is particularly relevant to this white paper, as 'store and forward' often happens at the enterprise boundary. For example, SendMail is used to route corporate e-mail to and from the internet – 90% of Fortune 1000 companies use the software on their networks. When an external e-mail sender looks up the MX record for Store & Co. in the above example, they may well be given sendmail.store.com, even though the recipients they want to send e-mail to are located on sfa.store.com and la.store.com. Receiving e-mail from outside an enterprise thus typically involves relaying steps, with the driver in part being to limit the internet exposure of the host of internal e-mail servers.

3.3 SMTP Vocabulary

Whilst internal mail delivery inside a company will now usually happen via proprietary protocols such as Microsoft MAPI or IBM Lotus NRPC (discussed later), the vast majority of inter-company e-mail is transferred via SMTP. The success of SMTP has been less to do with its long-term vision as a protocol, and more to do with being in the right place at the right time and riding the wave of internet adoption.

SMTP, now over 20 years old, can now be described as a legacy protocol, and is essentially described as such by its inventors. While the computing industry is currently pondering the shift from 32 bit to 64 bit computing, SMTP still operates in 7 bit (the eighth bit is reserved for parity).

7 bit is actually so low (2^7 gives 128 possibilities) that it can only accommodate Seven-Bit ASCII text, which does not support the international character set. And

herein lies what has been described as one of the 'ugliest hacks' remaining in the computer world. In order to push international character set messages and file attachments through SMTP, they must be encoded down to 7 bit. This is now done using the multipurpose internet mail extension (MIME) protocol, and depending on the method used to encode the content, the data volume is immediately increased between 10% and 50% with a typical figure (using base64) of 33%.

3.4 Retrieving and Storing E-mail

Local e-mail clients are typically not run on a 24/7 operations basis – users open the application (such as Microsoft Outlook) when they are using their laptops. This means that a user's incoming e-mail must be stored somewhere before (and possibly after) they access it. In addition, SMTP is not a suitable protocol for an e-mail client to communicate with an e-mail server, and thus a new protocol was needed to allow this.

The first mainstream protocol for retrieving email was Post Office Protocol (POP), invented in 1984, with POP3 being the version that has achieved mass adoption. Similar to SMTP, POP3 is a human-readable command/response protocol implemented over a reliable transport (typically TCP).

The open protocol intended to succeed POP3 was Internet Message Access Protocol, and IMAP4 – the current version – was created in 1996. IMAP4 has seen little enterprise adoption due to the overwhelming dominance of proprietary e-mail tools such as Microsoft Exchange and IBM Lotus Notes.

4 BlueSpace Overview

4.1 Introduction

BlueSpace is 'next generation' e-mail – T-Mail. Based on a truly relational architecture, it delivers infrastructure cost savings, business process integration and security, as discussed in depth in the BlueSpace *Relational Messaging White Paper*. BlueSpace T-Mail can be accessed through a proprietary client, through Microsoft Outlook and through Lotus Notes as well as through a web browser. A brief overview of the BlueSpace architecture follows, as knowledge of this architecture is essential to understanding the BlueSpace T-Mail cache.

4.2 BlueSpace Architecture

A simplified view of the core relational structure of T-Mail is presented in Figure 4.

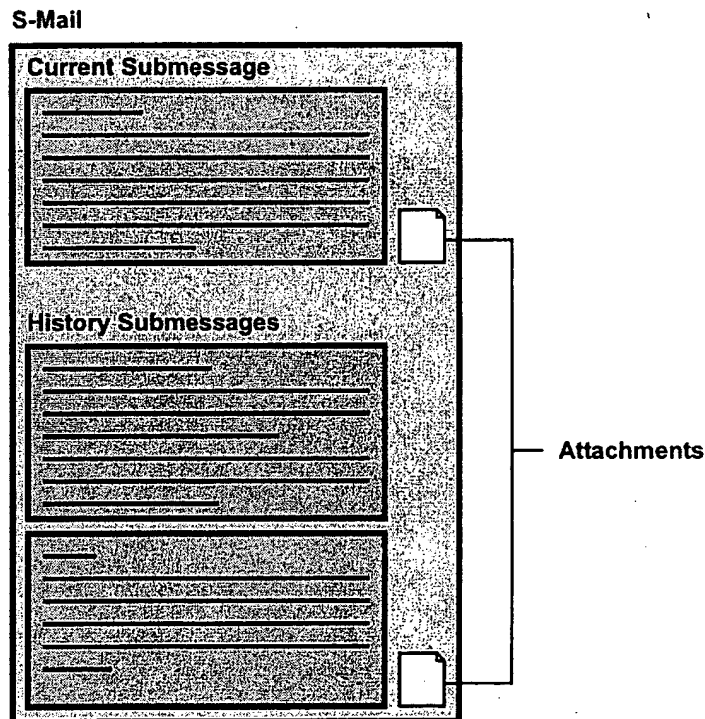


Figure 4 – T-Mail relational structure

A T-Mail can be thought of as a container with relational links. It does not contain content itself, but rather points to submessages in which content resides. When a user writes a message, they are actually writing a submessage, and then sending it inside a T-Mail to other users. A T-Mail has the current submessage (the one that the sender of the T-Mail has written specifically for the recipient) and then a number of history submessages, which contain the history of the communication exchange. The diagram in Figure 4 might reflect a user sending a draft document with comments to a second user, who replied, and then the first user replied with a second revised draft.

The following example (graphically represented in Figure 5) is intended to further explain the T-Mail to submessage relationship, and it will be revisited in Section

4.5 when discussing how BlueSpace T-Mail caching works, i.e. where information goes and the handshaking each 'actor' carries out:

- Alice, an employee of Store & Co in the New York office, sends a T-Mail to John, an employee working in the San Francisco office. It introduces a new policy on sales margins, and has an attachment outlining the new rules.
- Over a year later, John closes a very large sale, and attempts to understand the margin rules for complex, high-value sales. John does not read the attachment that Alice sent, assuming it's probably out of date anyway. Instead, John sends a reply to Alice asking how the rules will work, and copies in his boss – Peter (who also works out of the San Francisco office).
- Alice replies to both John and Peter, presenting an updated margin policy as an attachment.
- Peter reads the T-Mail from John, and then the T-Mail as well as its respective attachment from Alice.
- John then reads the T-Mail from Alice, but as he is displeased with the answer, he reads the original policy that Alice sent out as an attachment to her first T-Mail. Having found broad agreement between the two policies, he resides himself to less margin that he was hoping for.

From the above description, John and Peter both work in the same office (San Francisco) whereas Alice works in New York. The T-Mails transferred in the above example and their submessage components are visualised in Figure 5.

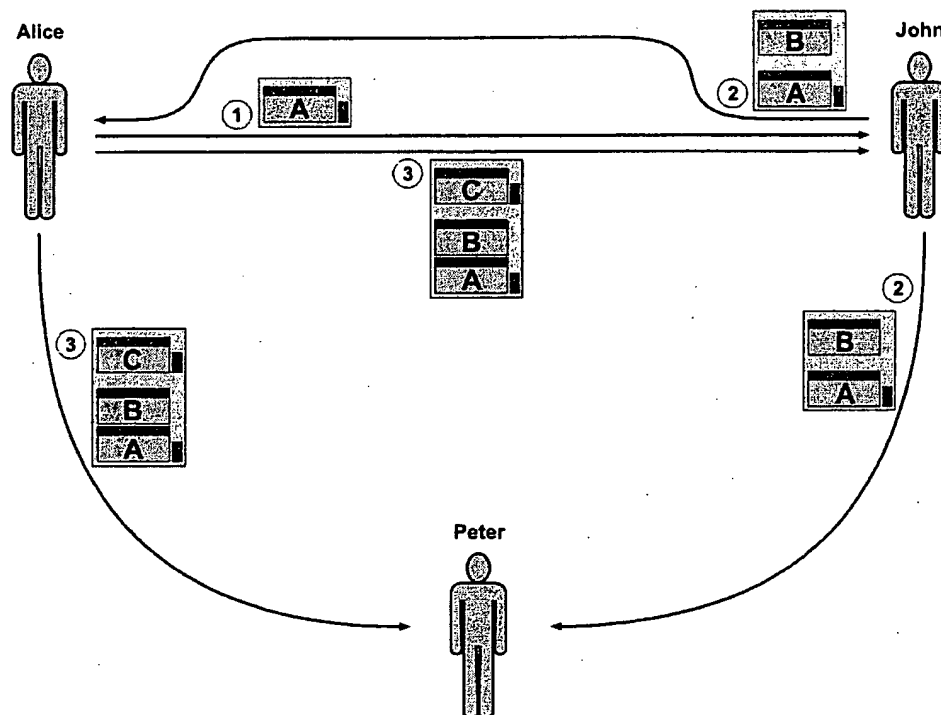


Figure 5 – Example of T-Mail flow

In step 1, Alice composes the first T-Mail with submessage A and attachment A' (represented by the dark blue square in Figure 5). John replies (copying in Peter) in step 2, creating the second T-Mail, which contains submessages B|A,A' (B

being the current submessage, whilst A and associated A' are the only history submessage). In step 3, Alice replies to both John and Peter, creating the third T-Mail, which contains submessages C,C'[B,A,A']. C is the new submessage written by Alice, and C' is its associated attachment, whilst B, A and A' are the history portions.

4.3 User Addressing

Inside BlueSpace instantiations, user addressing and directory services are not particularly critical – all users are local and therefore there is no need to route mail between BlueSpace servers.

In the context of this white paper external routing is a given, and thus BlueSpace identities and directory services must be outlined. A piece of feedback that has been continually received from BlueSpace customers and prospects has been the annoyance of having multiple identities for each user in their different (existing) internal systems. Most enterprises are attempting to reduce the number of user directories they hold, and certainly do not wish to add another. BlueSpace has listened to this feedback, and thus decided to build BlueSpace user identities on the existing SMTP addressing standard.

SMTP addressable e-mail identities have been used for three decades, and are by far the dominant inter-enterprise user addressing system. In general, SMTP works rather well for resolving a name such as john.smith@store.com to a real identity on a mail server somewhere on the internet. Such e-mail addresses are then often added in to an LDAP or Active Directory service that contains other information about the users.

The mechanisms of SMTP-based e-mail routing, using DNS MX records for domain names, were outlined in Section 3.2. In BlueSpace, a user types in the recipient's e-mail address into the 'To:' field, and then sends the mail. The BlueSpace server then uses an MX record lookup to determine how it should route the mail, as illustrated in Figure 6.

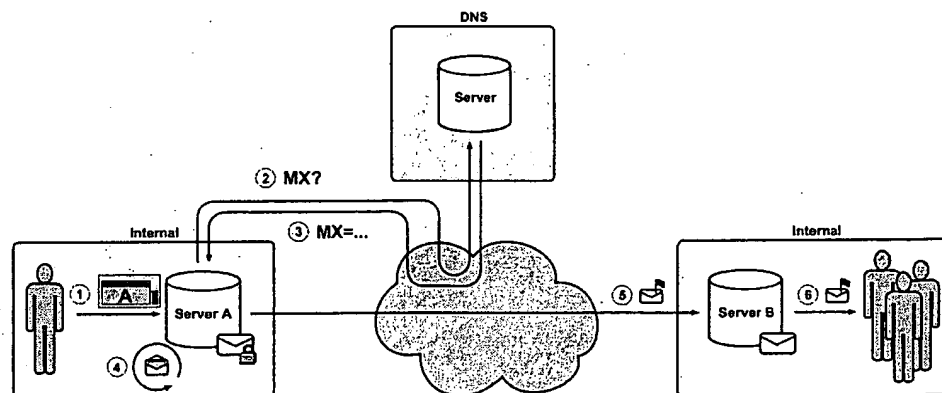


Figure 6 – MX lookup by BlueSpace to route e-mail

The steps in Figure 6 are as follows:

1. User sends T-Mail via BlueSpace server A to three external contacts with SMTP e-mail addresses – 'john.smith@store.com', 'susan.hindle@store.com' and 'bob.jones@store.com'.
2. Server A does a DNS MX lookup to find out the primary e-mail server for store.com

3. DNS server indicates that mail.store.com has the lowest MX value, and is thus the primary mail server for Store & Co.
4. BlueSpace wraps the T-Mail in an SMTP wrapper, making it into an e-mail that can be sent using normal SMTP.
5. BlueSpace sends the e-mail to server B using as SMTP connection.
6. The recipients – John Smith, Susan Hindle and Bob Jones – can access the e-mail from server B.

Note that steps 4 and 6 are simplified, as described in Section 5. A user communicating with a BlueSpace user carries out a similar MX lookup process.

4.4 Content Types

T-Mails actually contain two types of information:

- **Content data:** When a user sends a T-Mail, they are publishing a submessage. The submessage may be forwarded to other recipients, replied to, etc – but it will never be changed, and thus is suited to data warehousing. Content data is amenable to caching.
- **Audit data:** The audit data can be thought of as a distribution list for the published content. It tracks everything that happens to the content, such as which users read it, and which forward it on to other users. Audit data changes regularly, as a recipient may read a T-Mail at any time, so it is transactional in nature. Audit data is not suited to caching.

In Figure 5, the submessages A, B and C are all content data, as are their associated attachments; A' and C' (shown in dark blue). The cyan-coloured elements in Figure 5 represent the audit data – information such as whether the recipients have read the submessage or downloaded the attachment.

4.5 BlueSpace T-Mail Cache

BlueSpace T-Mail caching has been described in detail in the *BlueSpace Mail Caching* white paper, thus a summary is presented here to assist in constructing a foundation on which to build an explanation of BlueSpace T-Mail handling. Inspection of Figure 7 should reveal immediate similarity between the caching method deployed in BlueSpace and that used for web protocols.

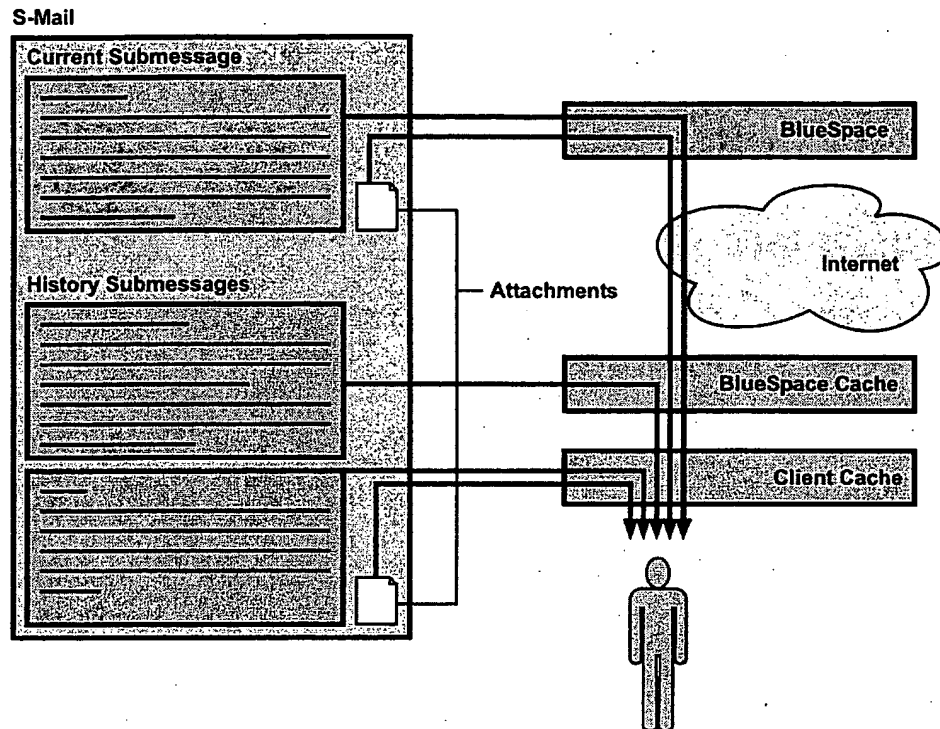


Figure 7 – BlueSpace T-Mail cache in operation

A T-Mail is purely a container that points at other resources, in the same way that a web page can have links to graphics. In the same way that an internet browser will first load the web page and then the necessary graphics, a T-Mail is loaded into the client first, followed by the submessages.

Submessage content data cannot change, and is thus highly suited to caching. It is also far larger in size than the other data type referenced by T-Mails – audit data. T-Mails reference submessages by URLs, and submessages are presented as XML documents to the client. The process of loading a T-Mail (illustrated in Figure 7) is thus:

- **T-Mail retrieval:** Retrieve the T-Mail container from the centralised server, and inspect the referenced URLs.
- **Submessage content data retrieval:** Retrieve the submessage content data from the client cache, or if not found there then from the BlueSpace cache, or if not found there then from the centralised server.
- **Submessage audit data retrieval:** This may change at any time, and thus must be retrieved directly from the server.

Note that once a submessage has been retrieved and has audit data associated with it, viewing the submessage again whilst online will cause a checksum confirmation with the central server thus determining whether the audit data in the client should be updated.

4.6 Impact on Infrastructure

Metrics and ROI calculations are available from BlueSpace giving precise figures on infrastructure savings, though these numbers will vary from organisation to

organisation depending on the design of the existing e-mail architecture and the organisation's e-mail behaviour. To summarise infrastructure impact briefly:

- **Servers:** BlueSpace is designed to be centralised, scaling up to 250,000 users in a single installation. This greatly reduces the number of servers required – a PC-based server can support up to 1000 users on current editions Microsoft Exchange, but can support 10000 users on BlueSpace. BlueSpace customers can expect in excess of a 70% decrease in server hardware.
- **Licensing:** As the number of servers increases, the licensing costs for software such as operating systems, mail, collaboration, and directory servers also increases. BlueSpace dramatically cuts down the number of separate software licenses required to run an e-mail system.
- **Storage:** Each submessage is stored once in a centralised database acting as a mail store, significantly reducing costs compared to other e-mail servers. In addition, mail is stored in a compressed format rather than a bloated format such as MIME. Storage savings vary from 60% to over 80% depending on the existing e-mail infrastructure (distributed mail systems use far more storage).
- **Backup & Archiving:** Many mail systems rely on repeated batch backups of the mail store, in which the entire mail content is archived to tape once each week or month and placed in long-term storage. BlueSpace negates the need for repeated batch backup of the mail store by managing content itself at the software layer. Batch backup can still be carried out, but its use can be limited to disaster recovery rather than long-term retention, thus the backup tapes can be cycled. Backup savings using BlueSpace are typically in excess of 90%.
- **Bandwidth:** BlueSpace has been designed to use the minimum possible bandwidth by having efficient protocols similar to web protocols, not e-mail protocols. Data presented below indicates the bandwidth savings generated by efficient protocols alone. The BlueSpace T-Mail caching model (detailed below) further reduces the network traffic attributable to mail.
- **Business Continuity:** E-mail is now a critical service. BlueSpace allows a high-availability mail system without expensive methods that rely on infrastructure 'hard wiring'. BlueSpace was designed to be a transactional system.
- **Management:** BlueSpace was designed to involve low maintenance by automating tasks such as backup and storage management, and using less servers which significantly lowers TCO, particularly when compared to distributed mail architectures.

The above areas are pure, immediately measurable infrastructure cost savings. In addition to these hard cost savings, BlueSpace also provides significant soft return on investment (ROI), by reducing business risk, structuring e-mail around business processes, and improving productivity.

4.7 Protocol-based Bandwidth Savings

BlueSpace Mail Transfer Protocol (BMTP) is far more efficient than existing protocols with respect to the amount of data transferred. This is reflected in the data shown in Table 1.

Action	Outlook 2000			Netscape Messenger 7.0		Blue-Space
	<i>MAPI</i>	<i>POP3 SMTP</i>	<i>IMAP4 SMTP</i>	<i>POP3 SMTP</i>	<i>IMAP4 SMTP</i>	<i>T-Mail</i>
Send a 1K plain text item	5.35 KB	9.34 KB	16.7 KB	7.62 KB	13.6 KB	1.41KB
Send a 2K plain text item	6.44 KB	16.1 KB	30.2 KB	13.8 KB	26.1 KB	2.03KB
Send a 1K plain text item with 10K attachment	21.1 KB	23.9 KB	45.7 KB	22.6 KB	43.2 KB	6.22KB
Send a 1K plain text item with 50K attachment	65.3 KB	84.1 KB	166 KB	84.2 KB	165 KB	25.5KB
Read a 1K plain text item	5.12 KB	8.69 KB	7.77 KB	8.31 KB	7.12 KB	1.41KB
Read a 2K plain text item	6.09 KB	15.4 KB	14.6 KB	15.0 KB	13.8 KB	2.03KB
Read a 1K plain text item with 10K attachment	10.1 KB	23.1 KB	7.77 KB	22.7 KB	7.12 KB	1.41KB
Open the attachment	11.9 KB	N/A	23.5 KB	N/A	15.1 KB	4.81KB
Read a 1K plain text item with 50K attachment	10.1 KB	82.8 KB	7.77 KB	82.1 KB	7.12 KB	1.41KB
Open the attachment	56.8 KB	N/A	83.2 KB	N/A	75.2 KB	24.1KB

Table 1 – Mail network traffic comparison

Note that the data in Table 1 for sending e-mail via POP & SMTP, and IMAP & SMTP both use SMTP as the protocol to send the e-mail to the server – the data traffic is different due to varying client/server handshaking.

4.8 Summary

T-Mails are relational containers that point to content data generated by users, and audit data reflecting users' actions. Content data is similar to publishing in that the content never changes and is thus suited to caching, whilst audit data changes regularly and is thus not suited to caching.

BlueSpace removes significant infrastructure and personnel costs through efficient delivery of an e-mail service. In addition, BlueSpace also reduces business risk and increases productivity by structuring the mail layer around business processes.

5 BlueSpace Discovery Service

5.1 Introduction

T-Mails are containers for submessages, as described above. The relational structure between T-Mails and submessages is key to the architecture of BlueSpace. Submessage content data can be thought of as being 'published', in the sense that it never changes once it has been written, and is at some point deleted. Submessage audit data changes regularly, as it records the actions of users in the mail system (such as reading and replying to T-Mails, thus changing the states of submessages and generating new submessages).

BlueSpace allows two methods of communicating between servers – SMTP and BMTP. As illustrated in Figure 8, a single mail can be sent simultaneously by both protocols if some of the recipients require the legacy SMTP method, whilst others support the BlueSpace BMTP method. The T-Mail is 'wrapped' in an SMTP envelope and then transferred through an SMTP gateway.

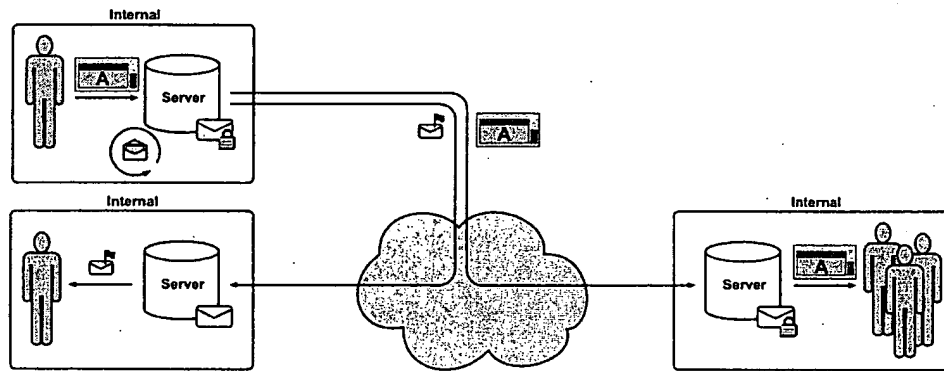


Figure 8 – Mail sent as both T-Mail and e-mail

There are three obvious questions regarding BlueSpace mail handling:

- Why would a corporation want to send T-Mails to other corporations rather than just sending e-mails?
- How does BlueSpace BMTP as used for sending T-Mails differ from SMTP, which is used for sending e-mails?
- How does a BlueSpace server know when to send an external message by e-mail and when to send it by T-Mail?

These three questions are now dealt with in turn.

5.2 T-Mail Superiority

Table 2 lists the ways in which T-Mail is superior for external mail when compared to conventional e-mail by SMTP.

<i>E-mail via SMTP</i>	<i>T-Mail via BMTP</i>
Transfer protocol inefficient. Attachments typically increase mail size by 33% during mail transfer due to base64 encoding (as	Transfer protocol efficient due to compression of mail before sending

described in Section 3.3)	
Entire mail always transferred even if receiving mail server already has a copy of most of the message (e.g. a reply chain)	Only the new submessage transferred to a receiving mail server that already has the history submessages
Relies on a 'store and forward' principle that may relay mail through intermediate mail servers	Mail always transferred directly between the sending and receiving mail servers
Mail delivery not guaranteed	Mail delivery guaranteed
Mail receipts based on trust mechanism that requires active confirmation by receiving user	Mail receipts enforced at the protocol level, transparent to the end user
Security requires PKI layering on top, which is typically complex and unusable by non-expert users, as it presumes end-user understanding and active involvement	Security handled at mail server level, transparent to the end user, and set up automatically (as described in detail in Section 5.4)
Risk of losing structure of T-Mail as message relayed purely as simple text string (e-mail itself has no rigorous message structuring)	Maintains T-Mail / submessage structure as BMTP 'understands' the way T-Mail works
Risk of losing business process information as modern usage of SMTP makes no explicit provision for additional descriptive fields, and meta data can be corrupted by mail clients, gateways and relays	Maintains business process information as BMTP was designed to preserve such information throughout mail chains
No reliable audit handshaking as SMTP does not make provision for reporting actions carried out by users	Reliable audit handshaking as BMTP makes special provision for reporting actions carried out by users

Table 2 – E-mail versus T-Mail for external mail

5.3 BMTP Versus SMTP

An extensive exposition on BlueSpace Mail Transport Protocol (BMTP) is presented in a dedicated white paper entitled 'BlueSpace Mail Transport Protocol (BMTP)'. For this paper, a brief comparison shall be made to SMTP, highlighting key BMTP approaches and principles.

As already described in Section 4.2, BlueSpace T-Mail relies on a relational messaging architecture that divorces T-Mail containers / envelopes from the submessage content written by users. This model was originally outlined in the seminal *Relational Messaging White Paper*. As the explanation of the model continued in Section 4.4, the concept of submessages being content data tracked by separate audit data was introduced. Content data was shown to never change, similar to a publication, and thus was suitable for caching. Audit data could change at any point, and was thus indicated to be unsuited to caching.

Section 4.5 overviewed the introduction of cache into the BlueSpace T-Mail architecture, as described in full in the *BlueSpace Mail Caching* white paper. The concept of a T-Mail being a relational container pointing at submessages was taken to the next level, by running hierarchical caching of submessages, thus reducing the data traffic generated through delivery of a mail service.

This difference (componentisation of mail) is carried into BMTP, allowing independent handling and fetching of mail constituents. BMTP also effects compatibility with SMTP-dependent mail systems such as Microsoft Exchange. BMTP also attempts to ensure that business process information and context is maintained at the enterprise edge when going through SMTP, using a variety of techniques including mail header modification and in-mail tagging.

For the purposes of this white paper, all that is required is a knowledge of the existence of BMTP, and a belief that using BTMP over SMTP is desirable when possible.

5.4 BlueSpace Discovery Service

There are two modes of operation of BlueSpace Discovery Service:

- **Pre-discovery:** In this mode, two BlueSpace T-Mail servers discover that they are both BMTP-capable before sending any mail between them, and thus initiate a BlueSpace handshake.
- **Post-discovery:** In this mode, one BlueSpace T-Mail server sends a conventional e-mail via SMTP to the other T-Mail server, which then discovers that the sending server is BMTP-capable, and initiates a BlueSpace handshake.

Both of these two modes are required, due to the topologies of many corporate mail systems and the mail 'store and forward' principle, as shall become apparent in Section 5.5.

5.4.1 Pre-discovery

The process of BlueSpace pre-discovery is depicted in Figure 9.

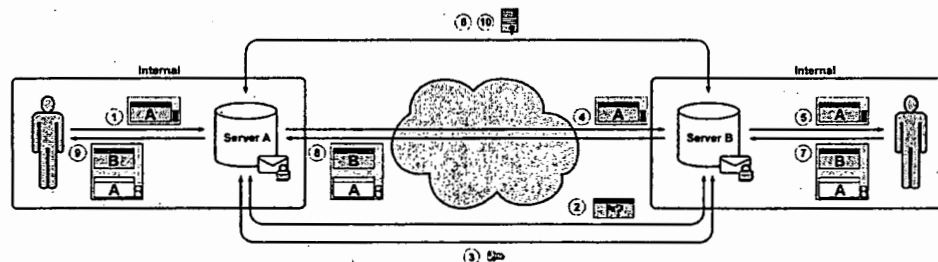


Figure 9 – The process of pre-discovery

The steps are as follows:

1. The sending user sends a T-Mail containing submessage A using their client, communicating with the sending BlueSpace server, Server A.
2. Server A initiates SMTP communication with the receiving server, Server B, and discovers that it is also a BlueSpace server.
3. Server A initiates a shift of protocol to BMTP, and conducts a key exchange (amongst other steps).
4. Server A transfers the T-Mail containing submessage A to Server B.
5. The recipient uses their client to view the T-Mail from Server B.
6. Server B conducts an audit handshake with Server A to indicate that the T-Mail has been read.

7. The recipient of the first T-Mail then replies, generating a new T-Mail, which itself contains submessage A (wrote by the user located on Server A) and submessage B (the new message).
8. Server B sends the second T-Mail to Server A.
9. The recipient of the second T-Mail uses their local client to access the T-Mail from Server A.
10. Server A conducts an audit handshake with Server B to indicate that the second T-Mail has been read.

Note that step 2 is described in detail in Section 5.7 and step 3 is greatly abridged – there are a number of separate steps involved.

5.4.2 Post-discovery

The process of BlueSpace post-discovery is depicted in Figure 10.

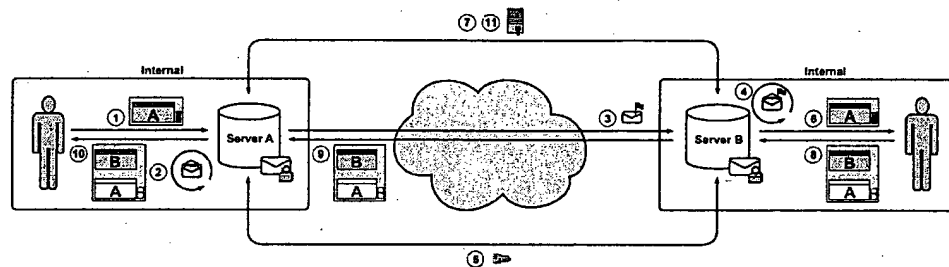


Figure 10 – The process of post-discovery

The steps (here abridged for simplicity as in Section 5.4.1) are as follows:

1. The sending user sends a T-Mail containing submessage A, communicating with the sending BlueSpace server, Server A.
2. Server A wraps the T-Mail in an SMTP envelope, i.e. makes it compatible with normal e-mail.
3. Server A sends the SMTP-wrapped T-Mail to Server B using the standard SMTP messaging protocol.
4. Server B unwraps the T-Mail, and should be able to recover the BlueSpace structure for the T-Mail.
5. Server B indicates to Server A that it is a BlueSpace server, initiating a shift of protocol to BMTP for all future mails, and conducts a key exchange (amongst other steps).
6. The recipient uses their client to view the T-Mail from Server B.
7. Server B conducts an audit handshake with Server A to indicate that the T-Mail has been read.
8. The recipient of the first T-Mail then replies, generating a new T-Mail, which itself contains submessage A (wrote by the user located on Server A) and submessage B (the new message).
9. Server B sends the second T-Mail to Server A.
10. The recipient of the second T-Mail uses their local client to access the T-Mail from Server A.
11. Server A conducts an audit handshake with Server B to indicate that the second T-Mail has been read.

5.5 Server Topologies

Figure 11 illustrates the three possible topologies for a BlueSpace T-Mail server communicating with another mail server.

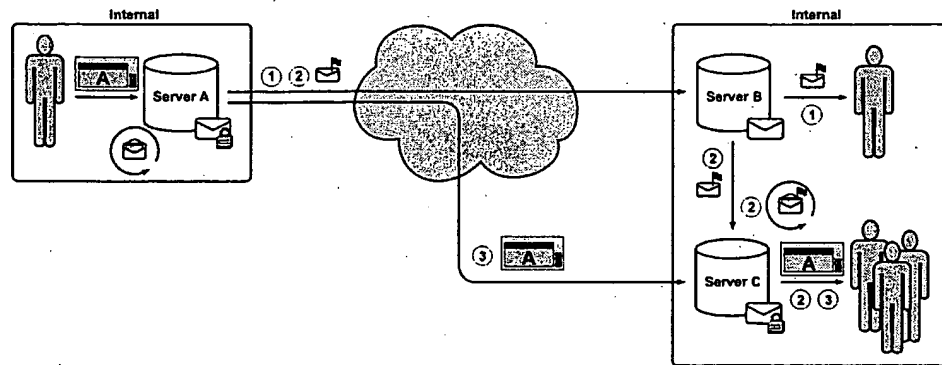


Figure 11 – Three topologies for external mail

The three topologies are as follows (each of which are independent and mutually exclusive):

1. In this scenario, a BlueSpace user on server A sends a mail to a conventional e-mail user hosted on a normal e-mail server, Server B, using SMTP.
2. In this scenario, a BlueSpace user on server A sends a mail to another BlueSpace user on Server C. Server C is positioned behind an SMTP gateway server, server B, and thus server A cannot connect directly with server C immediately. Server C is dependent on the 'store and forward' principle described in Section 3.1. Once server C has the mail, it unwraps the 'SMTP envelope' and presents it to the user as a T-Mail.
3. In this scenario, similar to previous scenario, a BlueSpace user on server A sends a mail to another BlueSpace user on server C. However, in this scenario, server A is able to connect directly to server C, and is not forced to go via server B. Server A thus sends the mail as a T-Mail directly to server C.

Scenario 1 is a normal T-Mail to e-mail gateway, scenario 2 must rely on post-discovery and scenario 3 can use pre-discovery. Mail routing is the key to the differences between scenarios 2 and 3 above, and mail routing relies on DNS MX records, as described in Section 3.2. In scenario 2, the sending BlueSpace server cannot communicate directly with the receiving server through an SMTP connection.

5.6 SMTP Syntax

BlueSpace Discovery Service in pre-discovery mode relies on receiving BlueSpace mail servers broadcasting their BMTP-compatibility through SMTP lines, thus SMTP must be discussed in depth. Lines are strings of data, passed backwards and forwards between two SMTP-compliant mail servers, the syntax being described fully in the relevant RFC and other literature.

A simple SMTP communication is shown in Table 3.

Party	SMTP line
-------	-----------

Sender	(opens network connection)
Receiver	220 store.com SMTP Service Ready
Sender	HELO hosta.hotmail.com
Receiver	250 store.com
Sender	MAIL FROM: <bob.johnson57821@hotmail.com>
Receiver	250 sender OK
Sender	RCPT TO: <john.smith@store.com>
Receiver	250 recipient OK
Sender	DATA
Receiver	354 Start mail input, end with <CLRF>.<CLRF>
Sender	Date: 22 Jan 1998 1245 PST From: Bob Johnson <bob.johnson57821@hotmail.com> To: <john.smith@store.com> Subject: New mail protocol I've found this great new mail protocol called BMTP. .
Receiver	250 OK
Sender	QUIT
Receiver	221 signing off
Sender	(drops network connection)

Table 3 – Simple SMTP server to server lines

In Table 3, Bob Johnson has sent an e-mail to John Smith, and the outgoing Hotmail SMTP mail server has established a connection with the incoming Store & Co. SMTP mail server to transfer the e-mail. In simple SMTP, each line of communication must be done in the logical order – each command (line transfer) must be acknowledged before moving on the next.

The original SMTP vocabulary was soon found to be limiting, and a series of extensions were introduced, such as RFC 1869 – Extended SMTP (ESMTP). Other extensions allowed a server to indicate the maximum size of mail it would receive, and pipelining, that allowed commands to be sent and acknowledged in batches rather than one by one

These new commands are broadcast by a receiving SMTP mail server when a server connects to it. Table 4 shows Extended SMTP lines, where the receiving server indicates to the sending server the various services it supports and (pipelining) and conditions it maintains on mail transfer (mail size maximum of 100,000 bytes).

Party	SMTP line
Sender	(opens network connection)
Receiver	220 store.com ESMTP Service Ready
Sender	EHLO hosta.hotmail.com
Receiver	250-store.com
Receiver	250-SIZE 100000

Receiver	250 PIPELINED
Sender	MAIL FROM: <bob.johnson57821@hotmail.com>
Receiver	250 sender OK

Table 4 – Extended SMTP server to server lines

As with all extensions to the original SMTP RFC, Extended SMTP was designed to be backwards compatible with older servers that do not support the newer extensions. The `HELO` command has been modified to a `EHLO` command, and if the receiving server responds 550 unknown command rather than 250-store.com, the sender switches to simple SMTP to carry out the mail transfer (and thus cannot take advantage of services such as pipelining).

5.7 BlueSpace Pre-discovery

BlueSpace Discovery Service operating in pre-discovery mode takes advantage of Extended SMTP to invent a new SMTP line: 250 `BMTP`, indicating support for the BlueSpace Mail Transport Protocol. In scenario 3 shown in Figure 11, two BlueSpace mail servers are communicating with each other for the first time.

The two servers are not yet aware that each of them is `BMTP`-compatible, thus server A must initiate the mail transfer as an SMTP connection to the receiving server, server C. Table 5 shows the SMTP traffic between server A, the sender, and server C, the receiver.

Party	SMTP line
Sender	(opens network connection)
Receiver	220 serverc.com ESMTP Service Ready
Sender	EHLO servera.com
Receiver	250-serverc.com
Receiver	250 <code>BMTP</code> ; Ports=7077,35; Version=1.1
Sender	<code>BMTP</code> recognised - attempting protocol switch
Receiver	250 sender OK

Table 5 – BlueSpace pre-discovery via SMTP lines

Server C indicates passes the line 250 `BMTP`; Ports=7077,35; Version=1.1, indicating that it is a `BTMP`-compatible mail server, that its preferred ports for `BMTP` connections are 7077 and 35 (in that priority order), and that it is using `BMTP` version 1.1.

Server A recognises the 250 `BMTP` command, and responds with the line `BMTP` recognised - attempting protocol switch. Server A then initiates a `BMTP` connection to server C, using whichever ports it has been allowed to connect through. The full syntax of `BMTP` is presented in a dedicated white paper entitled 'BlueSpace Mail Transport Protocol (`BMTP`)'. Having established a `BMTP` connection, server A is able to transfer the mail as a T-Mail.

If for any reason server A is unable to establish a `BMTP` connection with server C, server A does one of two sets of actions, depending on the administrator-determined server settings. Server A either reverts to a standard SMTP connection to transfer the mail, or it escalates the event to an administrator.

In scenarios 1 and 2 in Figure 11, server A is communicating with server B, which is not a BlueSpace server. In this case, server A operates as a normal SMTP server (supporting ESMTP), and transfers the mail as an e-mail. In a scenario in which a BlueSpace server is receiving e-mail from a conventional server, the BlueSpace server broadcasts its BSMTP compatibility. The sending server does not recognise the 250 BSMTP command, and so continues the SMTP connection as usual. The BlueSpace server then also continues the SMTP connection by behaving as a conventional ESMTP server, and thus receives the mail by e-mail.

5.8 BlueSpace Post-discovery

In scenario 2 shown in Figure 11, two BlueSpace mail servers are communicating with each other for the first time as per Section 5.7. In scenario 2, however, server A is forced to relay the first e-mail through server B before it arrives at server C. Due to the mechanisms of relaying and the use of MX record lookups for mail sent between enterprises, server A assumes that server B is the final location for the e-mail it has sent (i.e. it is unaware of the internal e-mail routing between server B and server C).

When server B transfers the mail to server C, server C uses the 250 BSMTP command, to indicate that it is a BSMTP-compatible server. However, as server B is a traditional mail server, the mail is relayed to server C by SMTP. When server C receives the mail, it discovers that it is in fact a T-Mail that has been wrapped in an SMTP envelope. Server C unwraps the T-Mail, as shown in Figure 11, and presents it to the recipient as a T-Mail.

Step 5 in Figure 10 indicates that when a BlueSpace mail server receives a T-Mail wrapped in an SMTP envelope, it then attempts to communicate directly with the originating mail server (as opposed to the relaying e-mail server). Thus in scenario 2 of Figure 11, server C conducts a BlueSpace handshake with server A, allowing a key exchange, amongst other steps. Server A and server C can then use BSMTP to communicate in the future – the mechanics of server memory are dealt with in Section 5.9.

One advantage of the SMTP protocol is that new headers can be added without changing the implementation of the servers that carry them. These headers will simply be passed on verbatim. Here is an example of the types of headers sent with an e-mail:

```
Return-Path: <sender@senderserver.com>
Received: from rcvhost (host.company.com [192.168.0.1])
    by mail.someplace.com (8.12.9/8.12.9) with ESMTP id hxx8793
    for <recipient@recipientco.com>; Mon, 10 Nov 2003 08:52:14 GMT
Received: from sendhost (host.isp.com [10.10.10.1])
    (authenticated)
    by prototype (8.11.6/8.11.6) with ESMTP id hAA8uU313195
    for <recipient@recipientco.com>; Mon, 10 Nov 2003 08:56:30 GMT
From: "Sender Person" <sender@senderserver.com>
To: "Recipient Person" <recipient@recipientco.com>
Subject: Example Subject
Date: Mon, 10 Oct 2003 08:54:08 -0000
Message-ID: <000801c3a768$343bd680$6500a8c0@sendhost>
X-Priority: 3 (Normal)
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook, Build 10
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2800.1165
Importance: Normal
```

Here, the sender was Microsoft Outlook. Note the inclusion of X-MSMail-Priority – a header used by Microsoft Mail clients only. In a similar way, the SMTP sending service for BlueSpace adds a set of headers:

```

Return-Path: <sender@senderserver.com>
Received: from recvhost (host.company.com [192.168.0.1])
  by mail.someplace.com (8.12.9/8.12.9) with ESMTP id hxx8793
  for <recipient@recipientco.com>; Mon, 10 Nov 2003 08:52:14 GMT
Received: from sendhost (host.isp.com [10.10.10.1])
  (authenticated)
  by prototype (8.11.6/8.11.6) with ESMTP id hAA8uU313195
  for <recipient@recipientco.com>; Mon, 10 Nov 2003 08:56:30 GMT
From: "Sender Person" <sender@senderserver.com>
To: "Recipient Person" <recipient@recipientco.com>
Subject: Example Subject
Date: Mon, 10 Oct 2003 08:54:08 -0000
Message-ID: <000801c3a768$343bd680$6500a8c0@sendhost>
X-BMTP-From: senderserver.com; 192.168.10.1
X-BMTP-Dir: senderserver.com; 192.168.10.2
X-BMTP-Data: JFDA29RJALSFAJB90283KSZLSF9210XJFKSLALJZNM283422ZQ
X-Priority: 3 (Normal)
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook, Build 10
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2800.1165
Importance: Normal

```

Note the addition of a set of X-BMTP header lines. These will be recognised by the final destination for that mail: a BlueSpace server. The destination server will recognise that the sender was a BlueSpace server (X-BMTP-From) and knows how to contact it directly (192.168.10.1). It also associates the senderserver.com domain with the indicated directory (X-BMTP-Dir: 192.168.10.2). Further information on the handling of the message is compressed into a X-BMTP-Data header. Further X-BMTP header directives are added as required, particularly when the sender requires special consideration when attempting to contact directly via BMTP, for example if a different port number is used.

The mail is then delivered to the user as normal. The server is then able to communicate back to the sending server using the details provided in X-BMTP-From, X-BMTP-Dir and X-BMTP-Data. If a direct path is available between the two servers, then BMTP can be used exclusively between them, without the need for SMTP wrapping. If, however, a direct path is not available, then SMTP-wrapping of T-Mails can be used instead of standard e-mails.

Further information on the tagging of T-Mails when wrapped in SMTP can be found in the BMTP white paper. The full syntax of BMTP is presented in a dedicated white paper entitled 'BlueSpace Mail Transport Protocol (BMTP)'.

5.9 Preventing Discovery Repetition

Another important feature of pre and post-discovery is that discovery happens once between two BlueSpace mail servers. Each BlueSpace mail server maintains a list of other BlueSpace mail servers with which it has communicated. After the pre and post-discovery processes, all future T-Mail transfers are initially attempted using BMTP. Should a BMTP connection fail, server A can retry, queue, escalate to an administrator and/or send the mail by SMTP, depending on server settings.

A server's list of other BlueSpace servers is subject to retention rules similar to cache engines, i.e. the lifetime of other servers in the list is a function of frequency of communication, time since last communication, remaining space for new entries etc. An administrator may often implement settings on a BlueSpace server such that in practise entries from the list are never deleted.

5.10 E-mail Architecture Problems

A typical e-mail architecture is shown in Figure 12.

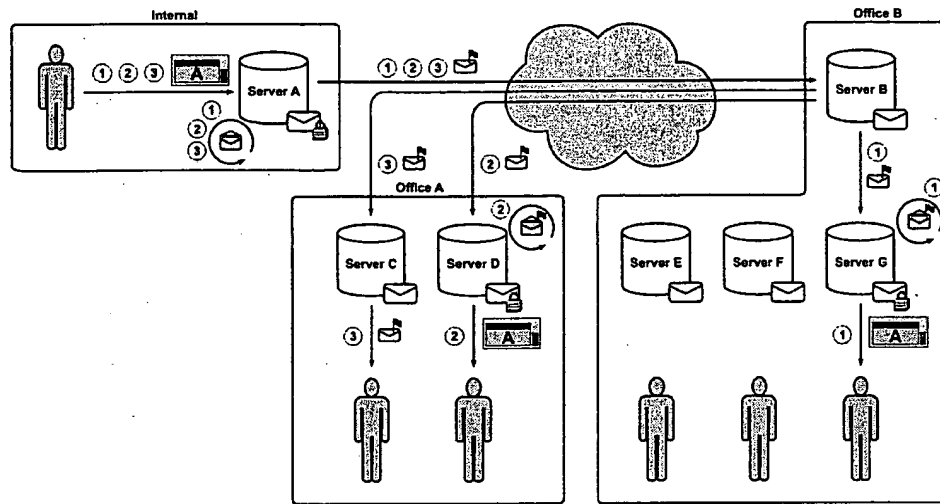


Figure 12 – SMTP gateway architecture

Whilst there is a continuing tendency to centralisation, most corporate e-mail architectures are still highly distributed, with hundreds and sometimes thousands of e-mail servers. Many corporate e-mail systems use an SMTP gateway such as SendMail to route e-mail to the internet – 90% of Fortune 1000 companies use the software on their networks, as highlighted in Section 3.2. Figure 12 is an example of such a case, where server B is operating as an SMTP gateway, forwarding mail to servers C and D in office A, and servers E, F and G in office B (where server B is also positioned). Servers C, E and F are conventional mail servers, whilst servers A, D and G are BlueSpace mail servers.

There is a problem with architecture in Figure 12, as illustrated in the following three scenarios:

1. In scenario 1, a user on server A sends a T-Mail to john.smith@store.com. A DNS MX lookup reveals server B as the main mail server, and thus server A connects via SMTP to server B, which in turn forwards the mail to server G where John Smith is located. Server G then initiates a BlueSpace handshake with server A, just as in the normal post-discovery service described in Section 5.4.2.
2. Scenario 2 follows on from scenario 1. In scenario 2, a user on server A sends a T-Mail to alice.hunter@store.com. Server A has already recorded the fact that the domain store.com has BlueSpace mail servers, but how can it know whether Alice Hunter is located on server G or not? Without any additional intelligence, server A is forced to send the mail to server B for forwarding (to server D). In fact, server A can never send a T-Mail directly to server G, as it is not aware privy to the correlation of users with mail servers inside Store & Co.
3. Scenario 3 also follows on from scenario 1. Again in scenario 3, a user sends a T-Mail to another Stores & Co. employee – don.harris@store.com. Server A is again forced to send the mail to server B, from which it is forwarded to server C – a conventional e-mail server.

Clearly this is not satisfactory, as given that most large corporate mail systems have the architecture shown in Figure 12, much of BlueSpace Discovery Service

would be rendered redundant. To handle these situations, BlueSpace has introduced directory interfacing.

5.11 BlueSpace Directory Interface

When a BlueSpace server has discovered another BlueSpace server, as in scenario 1 shown in Figure 12, each server stores the identity of the other, as already outlined in 0. When the sending server (server A in Figure 12) sends a second mail to the same domain (store.com) as per scenario 2 and 3, it first queries the BlueSpace mail servers already on its list for that domain, to ascertain as to whether the recipient resides on any of these BlueSpace mail servers.

In scenario 2 of Figure 12, the user (Alice Hunter) is not on server G (which John Smith was on), and thus server A sends the mail to server B from which it is forwarded to server D. Having unwrapped the T-Mail from the SMTP envelope, server D then carries out a BlueSpace handshake with server A, and thus server A adds server D to its list.

Various limits can be placed on the response of a BlueSpace mail server to ensure that it cannot undergo directory attack by a hacker. Whilst it is very unlikely that a spammer would attempt T-Mail spam, the positive returns might be used to indicate the users exist and are thus susceptible to normal e-mail spam at the same e-mail address. Specific safeguards include:

- An administrator can determine that a BlueSpace server replies to a maximum of, say, 50 user location requests per minute from a given IP address.
- A BlueSpace server can be made to tolerate up to a defined number of identity 'misses' on its own database per minute.
- A BlueSpace server can be made to tolerate up to a defined number of identity 'misses' on the directory of the enterprise as a whole (i.e. not just local to the server, but for all e-mail users in the enterprise).

As an additional fence for the would-be spammer, note that this service occurs through BMTP, and BMTP requires an initiating (licensed) BlueSpace server.

In the environment shown in Figure 12, Store & Co. has just two BlueSpace mail servers – other enterprises might deploy BlueSpace servers as replacements to conventional mail servers over a period of time. An enterprise could thus end up with tens or even hundreds of BlueSpace mail servers (if simply directly replacing legacy servers). An enterprise might thus decide that all internal BlueSpace servers should not add only their own address to an external BlueSpace server, but should rather provide the external server with a complete list of all current BlueSpace servers inside the enterprise.

This list transfer again occurs through BMTP, and an administrator is able to set various criteria before such an act is undertaken, e.g. a certain number of T-Mails are sent in the opposite direction back, i.e. from the administrator's own enterprise. Updating of these lists is subject to standard synchronisation techniques, such as periodic requesting of the current list of BlueSpace servers in a separate enterprise.

If a BlueSpace mail server is moved or consolidated, external BlueSpace mail traffic should be pointed at the new server identity (generally done by packet switching). As already indicated in Section 5.7, if the BlueSpace mail server is replaced with a conventional mail server, external mail servers will revert to SMTP to deliver the mail.

5.12 BlueSpace Mail Gateway Appliance

For some organisations, the concept of having lots of internal mail servers communicating directly with external mail servers may be somewhat unnerving, as it provides more potential entry points for a hacker to target. This is especially true in distributed e-mail architecture in which local mail servers are maintained by different departments, and security patching is not carried out on these distributed mail servers by the central IT function.

SMTP relaying partially resolves this problem, by providing a gateway between the internet and the internal network that can act similar to a firewall for SMTP traffic. The low functionality, high security, gateway SMTP server does not just engage in packet switching, but actually receives the e-mail, and then passes the mail on itself to its final destination.

The BlueSpace Mail Gateway appliance is designed to fulfil a similar function to SMTP gateways. The appliance is positioned in front of or in the place of the SMTP gateway server, and interfaces directly to the enterprise directory to ascertain as to whether email identities are BlueSpace mail accounts or conventional e-mail accounts.

The appliance is made the primary mail server in the enterprise's DNS MX record. Due to the nature of this role, the appliance can be load balanced, operate with a redundant fail-over, or can use an existing SMTP gateway as a fail-over (implemented using load balancing). The appliance can handle both incoming e-mail from external, conventional mail servers, and incoming T-Mail from external, BlueSpace mail servers.

E-mail handling by the BlueSpace Mail Gateway appliance is identical to that of a conventional e-mail gateway, as shown in Figure 13. The appliance receives an e-mail from an external e-mail server, finds the location of the recipient(s) by querying a directory service such as LDAP or Active Directory, and then relays the e-mail to another mail server inside the enterprise.

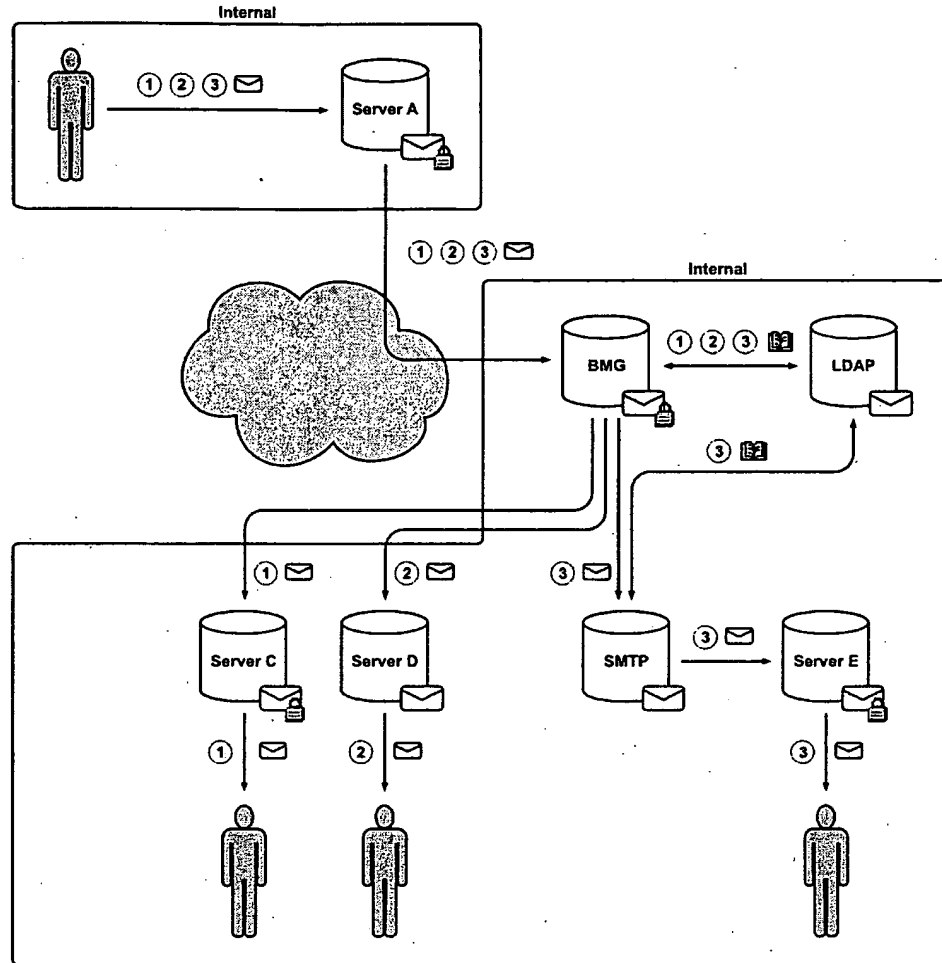


Figure 13 – BMG handling e-mail

There are three potential scenarios when the appliance forwards the e-mail to another server:

1. The BMG appliance relays the e-mail directly to a BlueSpace mail server on which the recipient is located.
2. The BMG appliance relays the e-mail directly to a conventional mail server on which the recipient is located.
3. The BMG appliance relays the e-mail to an SMTP gateway already present before the introduction of the appliance, which in turn relays the e-mail to a conventional e-mail server on which the recipient is located. The BMG appliance is effectively transparent in this case.

If an SMTP gateway is already present, the BMG appliance adds no real value in handling e-mail. Its value stems from handling T-Mails, as illustrated in Figure 14.

When handling incoming T-Mails, the BMG appliance always receives the incoming SMTP traffic, and converts the connection over to BlueSpace BMTP. Having done this, the appliance queries the directory to find out the final destination server, i.e. the server on which the account of the recipient is hosted. There are then four different scenarios for how the appliance behaves next, depending on its own settings and whether the final destination is a BlueSpace mail server or a conventional mail server.

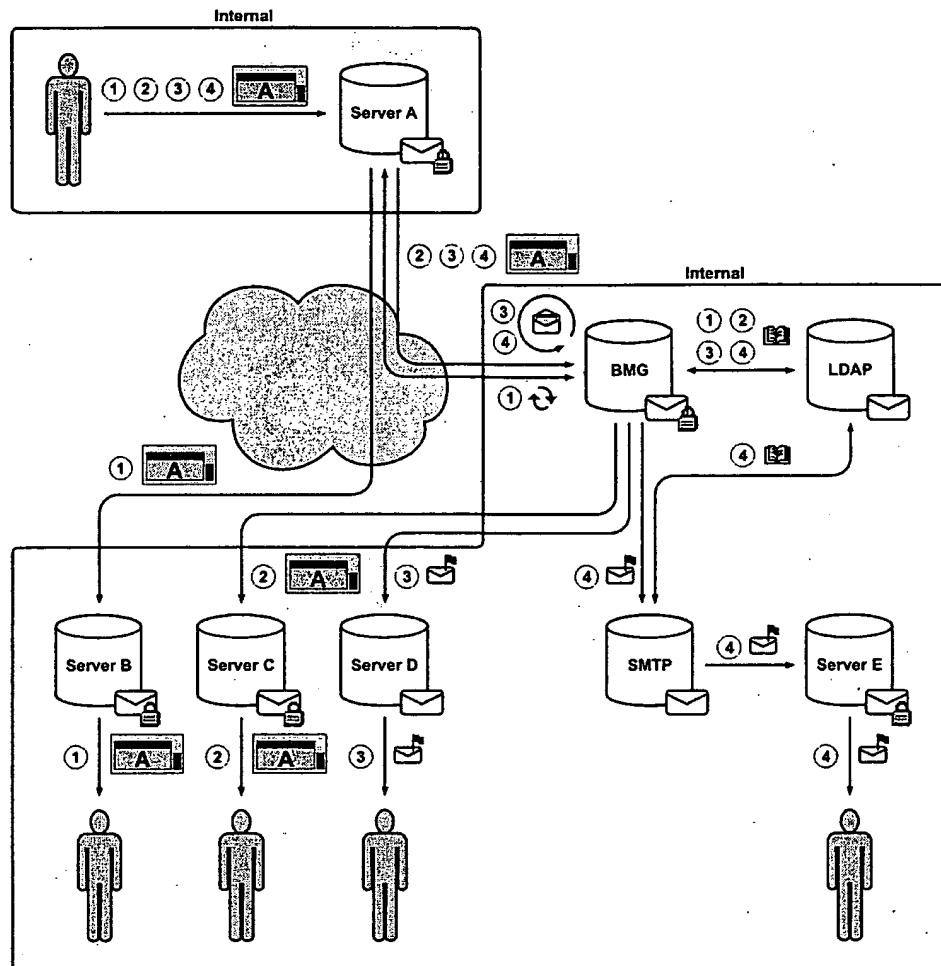


Figure 14 – BMG handling T-Mail

The four scenarios in Figure 14 are as follows:

1. The recipient is a BlueSpace user on server B. The BMG appliance has been instructed to forward BMTP connections to the destination server, and thus it provides server A with the address of server B. Server A then connects directly to server B, and transfers the T-Mail by BMTP. Future T-Mail transfers and audit handshaking happen directly between servers A and B.
2. The recipient is a BlueSpace user on server C. The BMG appliance has been instructed to receive T-Mails itself, and thus it receives the T-Mail from server A (acting as a T-Mail MTA) and then forwards it on to server C. Future T-Mail transfers and audit handshaking between servers A and B and mediated by server C.
3. The recipient is a conventional e-mail user on server D. The BMG appliance wraps the T-Mail in an SMTP envelope, and then forwards it to server D as an e-mail.
4. The recipient is a conventional e-mail user on server E. The BMG appliance has been set up to relay all conventional e-mail to the former SMTP gateway server, where it is processed by standard content filters and other such systems deployed by the enterprise. Having passed through the SMTP gateway server, it is then relayed to server E as usual.

There are a number of advantages in using a BlueSpace Mail Gateway appliance, over and above having an existing SMTP gateway in the mail system:

- All BlueSpace Discovery Service processes can be carried out in pre-discovery mode, as the BlueSpace Mail Gateway appliance is at the edge of the enterprise network, communicating directly with external BlueSpace servers. This ensures that all T-Mails are sent securely, not just all T-Mails after the first one for each external BlueSpace mail server (with post-discovery, the first T-Mail is wrapped in simple SMTP and cannot be encrypted, as detailed in Section 5.8).
- If BlueSpace mail servers are deployed in various geographical sites, especially if these sites have a degree of autonomy on security patching, allowing all internal BlueSpace mail servers to receive internet traffic creates multiple direct hacking risks. Having a single BlueSpace gateway to the internet, where all T-Mails are received by the appliance and then relayed on, reduces the number of externally peered BlueSpace mail devices to just one – the appliance itself. The central IT function can then ensure that the BMG appliance is fully patched. In addition the BMG appliance can have many services turned off compared to a BlueSpace mail server, as it is not hosting real user accounts.
- When the BlueSpace Mail Gateway appliance is acting as a BlueSpace T-Mail relay, an external T-Mail sent to users located on multiple internal BlueSpace mail servers is only transferred once across internet-peered bandwidth. The appliance receives the T-Mail and relays it as required, rather than having each internal BlueSpace mail server independently receive the T-Mail through the internet connection.

5.13 Internal Use

Whilst discussion of BlueSpace Discovery Service has until this point focussed on the enterprise border, i.e. communicating with external BlueSpace mail servers, BDS can also have an important role internally.

Many mail systems are developed in an ad-hoc manner, and there is no reason to believe that BlueSpace mail deployments inside enterprises will be any different to this historical precedent. In addition, BlueSpace will likely be subject to evolutionary deployments, where BlueSpace mail systems roll out across the enterprise over a period of time, replacing conventional mail servers. In this scenario it is highly likely that BlueSpace mail servers will be deployed independently by different parts of large organisations. The BlueSpace 'plug and play' mail server, aimed at smaller BlueSpace deployments, will also increase the chance of multiple BlueSpace mail servers inside a single enterprise.

BlueSpace Discovery Service allows BlueSpace mail servers that have been implemented independently to discover each other and switch to BMTP. This provides the security, network bandwidth savings, audit guarantees and business process information preservation discussed in Section 5.2.

5.14 Summary

BlueSpace Mail Transport Protocol allows BlueSpace mail servers to be compatible with existing SMTP servers, such as Microsoft Exchange, whilst also leveraging the relational structure of BlueSpace to provide tangible benefits when

communicating with other BlueSpace mail servers. These benefits include security, efficiency, audit guarantees and context preservation.

BlueSpace Discovery Service empowers BlueSpace mail servers with the ability to automatically switch between SMTP and BMTP dependent on the other party in the mail transaction. Whilst an administrator can determine settings for a BlueSpace mail server in how BDS operates, no administrator interaction is required in the day-to-day operations of BDS.

The BlueSpace Mail Gateway appliance is specifically designed for larger enterprise networks with complex existing mail infrastructures and a likelihood of multiple, independent BlueSpace mail server deployments. A BMG appliance allows pre-discovery for all external T-Mails, can reduce internet-peered mail traffic and can reduce the risk of mail servers being hacked by acting as a single internet-peered connection point for an enterprise's BlueSpace mail system.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

PATENT APPLICATION SERIAL NO _____

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE
FEE RECORD SHEET

06/28/2005 HDESTA1 00000027 60693605

01 FC:2005

100.00 DP

PTO-1556

(5/87)