IETF fax WG
Internet draft

G. Klyne, MIMEsweeper Group
D. Crocker, Brandenburg Consulting
5 November 2001
Expires: May 2002

Timely Completion for Internet Messaging Services

<draft-ietf-fax-timely-delivery-05.txt>

Status of this memo

Copyright Notice

Abstract

This specification provides a way to request timely completion for Internet mail delivery, for services such as facsimile and voice messaging. Traditional Internet mail uses a _postal_ mail model, with normal delivery having an indeterminate gap between delivery into a mailbox and processing by the recipient. Timely completion adds a timelines service feature and extends delivery processing all the way to the recipient. This specification provides a deterministic service quality response, while preserving most of

the traditional roles and responsibilities of the agents involved in email transfers.

It is essentially a profile of the Delivery Service Notification (DSN) and DELIVERBY extensions for ESMTP, along with a new TIMELY option to DELIVERBY with a new deterministic service quality response.

Discussion of this document

Please send comments to: <ietf-fax@imc.org>.

To subscribe: send a message with the body 'subscribe' to <ietf-fax-request@imc.org>. The mailing list archive is at <http://www.imc.org/ietf-fax/>.

Table of contents

1. Introduction

Traditional Internet mail uses a _postal_ mail model, with normal delivery placing a message into the recipient's mailbox and the recipient retrieving and processing the message sometime later. For traditional mail, delivery responsibility stops at the mailbox. However some uses of messaging require a service model that confirms receipt by the actual recipient, not just delivery into their mailbox. Also, some uses require delivery within a specified period of time.

Timely completion adds this timeliness service feature and extends delivery processing all the way to the recipient. This specification provides a deterministic service quality response, while preserving most of the traditional roles and responsibilities of the agents involved in email transfers.

RFC 1891 [4] defines an ESMTP extension for Delivery Service Notification (DSN), and RFC 2852 [5] defines one for requesting delivery of a message within a given interval. Timely Completion is essentially a profile of the DSN and DELIVERBY extensions for ESMTP, along with a new TIMELY option to DELIVERBY with a new deterministic service quality response This memo describes how to use those specifications, along with some small extensions, to achieve timely completion of message delivery.

1.1 Structure of this document

Section 2 gives the background, principal ideas and goals of this specification.

Section 3 describes the mechanisms used, and how they are combined to achieve the timely delivery goals.

Section 4 describes an addition to the ESMTP "Deliver by" extension which is one of the mechanisms used to achieve timely delivery.

Section 5 describes extensions to the DSN reporting format and status codes used to report conditions related to timely delivery requests.

Section 6 contains some non-normative discussion of implementation issues related to this specification.

Section 7 contains some examples uses of this specification.

1.2 Document conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [11].

> NOTE: Comments like this provide additional nonessential information about the rationale behind this document. Such information is not needed for building a conformant implementation, but may help those who wish to understand the design in greater depth.

1.3 Terminology

Delivery
: is the process performed by a delivery MTA in attempt to achieve final receipt of a message.

Final receipt
: means a message is accepted by a receiving agent, and is immediately available for disposition (i.e. received by a disposing agent). For example, if mail is received using a protocol like POP or IMAP, final receipt is not deemed to have occurred until the message has been transferred to the recipient's system. The difference between delivery and final receipt is due to the ability have an MTA store a message in a user mailbox, but not be able to notify recipient software of the delivery. Hence there can be arbitrary delay between the time the message is delivered into the mailbox and recipient user's software agent does any processing.

Disposition
: is receipt and processing of a message by a recipient. Timely notification of disposition is problematic to achieve because it is not always possible to determine that disposition has occurred, and in any case may be undesirable for privacy reasons.

Best effort
: indicates that a system will ensure that an assigned task is successfully completed under all but the most catastrophic of failure circumstances. Common failure modes, such as power failures, SHOULD NOT prevent eventual completion of a task.

Reasonable effort
: indicates that while a system will try to complete an assigned task, it MAY also indulge in behaviours, or make operational decisions, that significantly reduce the certainty of an

action's being completed in the face of disruptive circumstances.

Delivery interval
: a period of time, measured in seconds, allowed for completion of message delivery and/or receipt.

Timely delivery
: means a message is delivered to a recipient's MTA within a specified time interval.

Timely delivery
: means a message is delivered by a recipient's mail transfer agent (MTA) -_ that is, typically, handed to the user's mailbox -- within a specified time interval.

Timely receipt
: means final receipt of a message, by active recipient user software, within a specified time interval.

Timely completion
: means that notification of a requested timely receipt is received by the sender of a message within a determined time interval. Non-receipt of notification within that interval is indicative of failure.

MTA, Mail Transfer Agent
: is an email system component with the roles of receiving, transferring, and delivering messages.

MUA, Mail User Agent, disposing agent
: is an email system component with the role of preparing and sending, and/or receiving and processing, messages. MUAs are the endpoints between which emails are sent; MTAs are relays on the path between a sending MUA and a receiving MUA.

Delivery MTA
: is the final MTA in an MTA relay sequence; it accepts a message and passes it to the receiving MUA.

2. Background and goals

RFC 2852 [5] provides a mechanism to request timely delivery of a message using SMTP. While this is helpful, it falls short for some usage profiles, such as timely processing of fax messages. These profiles are determined, in part, by the capabilities of traditional facsimile [8].

2.1 Background

Traditional email [2] is open-loop. The sender of a message normally has no certainty if or when a message is delivered. (A separate memo [6] contains a discussion of some open- and closed-loop issues in email.)

To be more than just a hint to the message transfer system, timely completion requires a deterministic confirmation mechanism, to close the loop. This is provided by DSN [4].

Some different levels of timeliness can be identified:

(a) timely delivery to the recipient's MTA

(b) timely final receipt by the recipient's disposing agent

(c) timely disposition (receipt and processing) by the recipient

(d) timely notification to the sender of delivery

(e) timely notification to the sender of final receipt

(f) timely notification to the sender of disposition by the recipient's user agent

From the sender's point of view, timely confirmation of disposition is the most desirable requirement. As noted previously this can be problematic, but timely notification of final receipt is practically as useful.

2.2 Basis for timely completion

A premise of the service specified here is that timeliness CAN be achieved using existing protocols, with appropriate software design and operational management. But the sender and receiver do not control all the relays used:

- The real issue is lack of determinism: a message might be delivered quickly, or it might take hours or even days, or it might not be delivered at all; the sender has little knowledge and no control.

- A second issue is post-delivery handling: will the recipient's user agent receive the message in timely fashion?

Then, assuming that the infrastructure is generally capable of achieving the desired timely completion, the main thrust of this

memo is provide protocol enhancements that put the sender in

complete control on those occasions when timeliness is not achieved.

One challenge to achieving this is dealing with uncertain transit times of the confirmation message over the return path (which is not necessarily the same as the forward path).

2.2.1 The SMTP "contract"

On accepting a message, a normal SMTP message transfer agent (MTA) accepts responsibility to:

(a) Use a best effort to ensure ultimate delivery of the message, or

(b) Attempt to provide notification that delivery could not be achieved.

This memo introduces mechanisms to allow this contract to be modified. A timely-completion MTA accepts responsibility to:

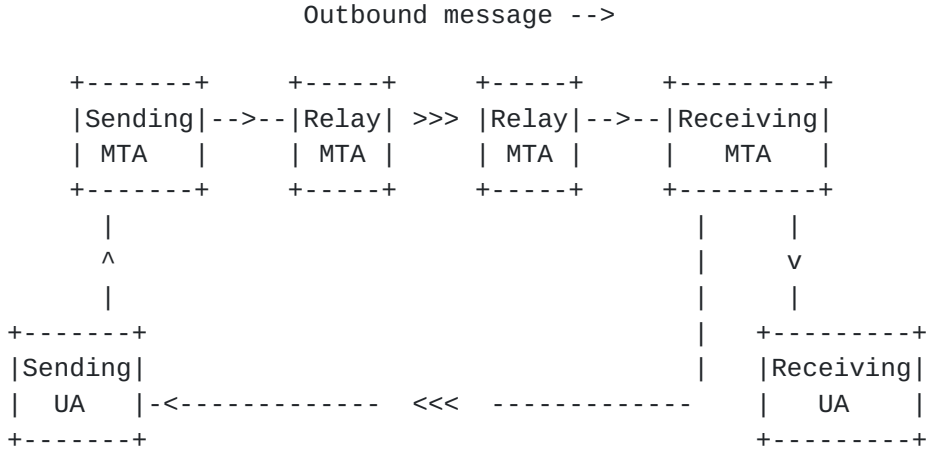
(a) Use a reasonable effort to ensure delivery within a specified time, and to provide timely confirmation of this, or

(b) Provide timely notification that delivery was not achieved as requested.

The sender can then decide a recovery strategy

2.2.2 Framework for timely delivery

The diagram shows typical SMTP message delivery and delivery status notification (DSN) paths. (The confirmation path is not necessarily the same as the message delivery path.)



<-- Return confirmation (disposing
agent)

As well as requesting timely delivery of a message, this specification needs to take account of the possibly varying characteristics of relays of the outbound and return message paths. Practically, it is possible to require that every relay on the outbound path recognizes timely completion semantics (using the ESMTP extension framework), but it is not possible to require this of every relay on the return path. Thus, it may be necessary to make some assumptions about the confirmation return path.

> NOTE: The uncertainty about return path characteristics might be removed by requiring an MTA to send any timely delivery notification to the MTA from which it was received, but this goes against trends in SMTP design and deployment. This might also raise state maintenance and hence scalability concerns.

The other issue apparent from the diagram is that providing timely delivery through the SMTP message relays does not ensure that the receiving UA will receive the message in a timely fashion. If the receiving MTA delivers to a POP mailbox, there is currently no way that it can guarantee timely receipt by the disposing agent.

2.3 What does the TIMELY option add?

The TIMELY option adds three elements to the DELIVERBY extension:

- It modifies the SMTP contract, permitting an MTA to commute its responsibility for delivering the message from "best effort" to "reasonable effort", with notification of outcome.

- It extends the reach of the timeliness constraint to cover final receipt: i.e. hand-off to a disposing agent (see below).

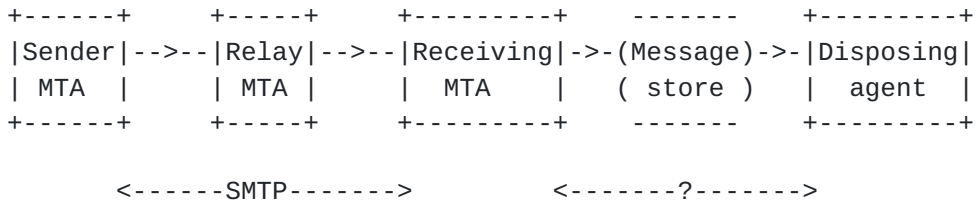
- It establishes a basis for determining the allowable time interval for certain behaviours initiated by the receiver of a message (i.e. DELIVERBY interval for delivery status responses, and how long to wait for possible duplicate message transmissions).

This is all consistent with the fundamental strategy of seeking to give the sender control over the whole message transfer process: if an MTA or the receiving agent cannot communicate the required guarantee, delivery is not completed and the sender is duly notified.

2.3.1 DELIVERBY and timely delivery

Timely completion of delivery is handled by the DELIVERBY ESMTP extension base specification [5]. However its scope ends with final delivery by SMTP, not covering final receipt by the disposing agent. The TIMELY option modifies the DELIVERBY semantics to cover the additional step needed for the message to reach the recipient.

Consider the following scenario:



The base DELIVERBY specification concerns itself with only participants in the SMTP transfers. But for the purposes of timely completion of final receipt, the sender must be able to specify the timeliness constraint to include this extra step.

The TIMELY option requires that the receiving MTA communicate with the disposing agent (in some unspecified way), and that it confirm final delivery of the message only if the disposing agent confirms that it will deal with the message in timely fashion, or that it will return an indication to the receiving MTA if it fails to do so. Simply putting the message into a POP mailbox would not meet this criterion.

2.4 Goals for timely completion

The primary goal is to allow consenting parties to establish a relationship that carries a guarantee of final receipt within a specified time, or timely notification that it was not achieved.

Further goals:

- Provide "while-you-wait" delivery of messages by email, where available infrastructure and connectivity permit.

- Deterministic behaviour, whereby sender who requests timely completion is able to determine with reasonable certainty, and in reasonable time, whether that request was successful.

- If the message cannot be delivered as requested, it SHOULD NOT be delivered at all. This means that a sender can choose other strategies for message delivery (e.g. if timely delivery by email

does not succeed, to resend the message as a traditional

facsimile; in such circumstances it is preferable that multiple copies of the message are not delivered).

- Operate within the existing ESMTP extension framework [3], using existing facilities where available.

3. Mechanisms for timely completion

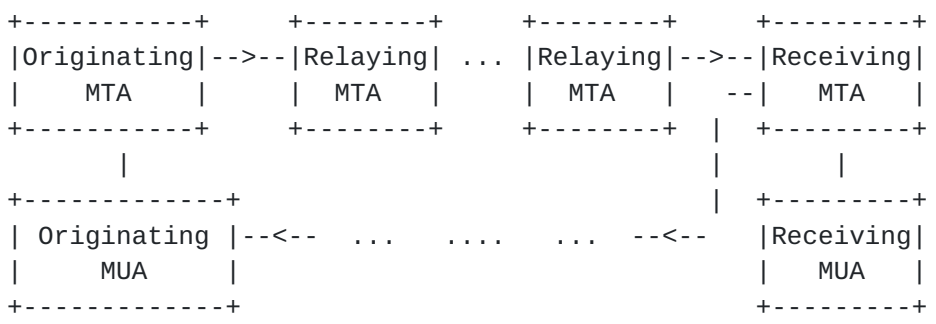
Deterministic timely completion is achieved through a number of ESMTP extensions used in concert:

- Delivery Status Notification ("DSN"), per RFC 1891 [4].

- Deliver-by ("DELIVERBY"), per RFC 2852 [5]

- A new TIMELY extension to DELIVERBY, that serves to modify the SMTP contract and also to establish that the receiving user agent can process the message in timely fashion as required, or provide timely notification of its failure to do so

The confirmation loop for successful delivery looks something like this:



The path through MTAs taken by the confirmation response is not defined, and may be different than the forward path of the original message.

3.1 Transmitting a message for timely completion

A transmitted message for which timely completion is required MUST include the following:

- An 'ENVID' parameter on the MAIL FROM command, per DSN [4]

- An 'ORCPT' parameter on the corresponding RCPT TO command(s), per DSN [4]. (This is to allow the sender to tell exactly which recipients were successfully delivered.)

- A 'NOTIFY=SUCCESS,FAILURE' parameter on the corresponding RCPT TO command(s), per DSN [4]

- A 'BY' parameter on the MAIL FROM command, per [5], with a 'by-mode' value of 'R'.

- A 'TIMELY' parameter on the MAIL FROM command, as described below, initially having the same time interval as specified for 'BY'.

The message MUST NOT be transmitted to any MTA that does not indicate support for all of these extensions, in its response to the EHLO command. In this case, a negative delivery status report MUST be generated, which SHOULD indicate the non-compliant MTA, the extensions that it does not support, and the name of the reporting MTA (per DSN, using the non-compliance reporting extensions noted later).

Standard DNS MX-based message routing, per RFC 974, SHOULD be used when sending or relaying the message.

> NOTE: Any strategies that vary standard MX routing should be used with care, and only with the goal of improving network transit times and timing consistency. These comments about mail routing apply especially to the handling of DSN responses.
>
> Ideally, there will be no intermediate relay between the sending and receiving MTAs, and in any case the number of such relays should be minimized to reduce timing variability on the transfer path.

3.2 Relaying a message

An MTA that relays a message for timely completion MUST support all of the ESMTP extensions noted above; otherwise it MUST NOT be given the message in the first place. When a relaying MTA accepts a message (by its 2xx status response to receipt of the message data), it becomes responsible for its onward delivery, including satisfying all of the options associated with the message.

In order to relay such a message, an MTA MUST note when the message was received, and the time when the attempt to transmit the message to the next MTA is initiated, and reduce accordingly the time interval used for the BY parameter. (The time interval SHOULD be taken to start with receipt of the MAIL FROM command.)

If the DELIVERBY time interval is reduced to zero or less (or less

than some system-configurable value indicating that delivery within the indicated interval is unlikely to be achieved) then the message

MUST NOT be relayed. Instead, a negative delivery status report MUST be generated indicating that the time for delivery of the message has expired, and the reporting MTA (per DSN, using the deliver-by extensions and/or non-compliance reporting extensions noted below).

The above behaviour is as specified for the DELIVERBY ESMTP extension; see RFC 2852 [5] for a definitive description of how to handle relaying of such messages. The following additional considerations are applicable when the TIMELY option is used:

The TIMELY parameter in the MAIL FROM command of a message in transit is copied unchanged when the message is retransmitted. Thus, any originally specified time interval is conveyed to the final MTA, to be used as a basis for selecting a delivery interval for returning a timely notification.

Standard DNS MX-based message routing, per RFC 974, SHOULD be used when relaying the message. (See note at end of previous section.)

If the first attempt to relay a message fails, the relaying MTA MAY assume that delivery within the desired time will not be achieved, and immediately indicate a delivery failure, indicating the name of the next-hop MTA. Alternatively, the relaying MTA MAY wait and retry the transmission, provided that the retry attempt will be performed within the remaining delivery period; if the transmission cannot be completed after one or more such retries then a negative DSN MUST be generated as noted above.

Any negative DSN generated SHOULD indicate the number of retries attempted (where 0 means no retries).

The choice to retry or not to retry is installation dependent. Effectively, when a relay does not retry, any responsibility for overcoming the delivery failure is passed back to the original sender. This strategy may be appropriate for cases where very rapid delivery is required or expected.

> NOTE: The presence of a 'TIMELY' option may cause a relay to abandon a message that it would otherwise retry (even given a 'by-mode' value of 'R'). One purpose of this option is to establish that responsiveness to the sender is more important than getting the message through. An effect of this may be to severely constrain the number and frequency of retry attempts.

3.3 Delivery MTA message acceptance

The MTA that performs final delivery of a message has responsibility for passing the message to a Mail User Agent. The exact mechanism by which this is achieved is a local matter, and is not defined here or by the Internet mail specifications. The delivery MTA, or its agent, is also responsible for generating any successful DSN message.

Before generating a success DSN message, the final MTA MUST ensure that all of the conditions for timely completion of the message have been achieved. Specifically, when the TIMELY option is used, it MUST ensure that final delivery to the disposing agent will be completed within the delivery interval indicated as the value of the BY parameter of the received MAIL FROM command.

The time interval for completion of final receipt SHOULD be taken to start with receipt of the MAIL FROM command.

> NOTE: Final receipt by an MUA is expected to include some guarantee of timely processing. Exactly what this constitutes may depend on the circumstances: in a simple case, depositing the message in a local mailbox and immediately notifying the recipient possibly constitutes final receipt. A more complex case would be that of a fax offramp, where final receipt may be completion of a successful outdial and transmission of the fax.

3.3.1 Timing of final receipt

In the presence of a TIMELY option, final receipt SHOULD NOT be indicated unless the delivery MTA can establish that the receiving MUA will deal with the message promptly. Here "promptly" means a reasonable waiting time for a human; e.g. that the message (or at least the start of the message) will be available to its intended final recipient within a period of, say, 30 seconds.

The relationship between the delivery MTA and receiving MUA can work in one of two ways:

- The MUA always processes the message promptly, barring exceptional circumstances. Queuing a message to a network printer would constitute such processing -- normally the message will be printed within seconds, even though it might be delayed if the printer runs out of paper. The delivery MTA can generate the final DSN when the MUA has accepted the message.

- The MUA attempts to process the message promptly and reports the

outcome within the remaining DELIVERBY period. If processing is not performed within the stated period, the message is abandoned

and failure is signalled back to the delivery MTA. The delivery MTA must hold off generating the final DSN until the MUA has provided a status report; if no such report is provided within the remaining DELIVERBY interval, it SHOULD report failure.

3.4 Reporting failures

When a relay or receiving MTA determines that a message cannot be delivered as requested to any recipient, a DSN report MUST BE sent back to the sender.

The following status codes indicated that message delivery has been abandoned, are used with DSN "Action: failed" for reporting conditions that are specific to timely delivery:

4.4.7: Delivery time expired -- failed.
Message delivery could not be completed within the specified time interval. This code is also used when the final MTA has accepted the message but has been unable to achieve final receipt within the requested interval.

5.4.9: Protocol required for timely delivery not supported.
A relay MTA was encountered that did not support the range of capabilities required for timely completion. Defined by [15].

4.4.1: Next MTA not accepting messages.
A relay MTA has been unable to contact a next-hop MTA, and has decided to abandon delivery. (See note in section 3.2 about a relay's options with respect to retries.) This code SHOULD be accompanied by a 'Retry-Count' DSN field.

4.3.3: Receiving MTA cannot honour required timely receipt.
A message has been delivered to a receiving MTA within the required delivery interval, but that MTA is unable to ensure timely receipt or timely notification of failure to do so.

The following status code is used with DSN "Action: delayed" for reporting delayed message receipt following delivery:

4.4.7: Delivery time expired -- continuing.
The message has been received by the delivery MTA and is in the process of final delivery, but that final delivery has not yet been completed.

A subsequent DSN SHOULD be sent when the final delivery succeeds or fails.

This status code is defined for situations where a receiving MTA has handed off the message to another agent for final delivery, and has therefore committed to provide a timely confirmation response, but the delivery agent has not signalled completion. For example, a fax dial-out gateway may have been invoked assuming that the outdial leg would complete within a given period, but has failed to do so.

3.5 Timely confirmation

Fully deterministic behaviour requires that the round-trip time to deliver a message and receive a confirmation response be within a known time interval.

As noted above, it cannot be guaranteed that confirmation of delivery or non-delivery will be transferred in timely fashion, although return path transit times often are comparable with forward path times. Use of the DELIVERBY extension for a message confirmation MAY serve to expedite its forwarding (noting that this is not a required behaviour; see implementation notes section 6.1 for further discussion).

Further, it is likely that perfect determinism can never be achieved using SMTP; e.g. see RFC 1047 [9]. Repeat deliveries are considered less harmful than lost messages, but even these should be minimized.

The following behaviour is followed to achieve near-deterministic timely confirmation:

- Always fail forward delivery if a non-TIMELY MTA is encountered.
- A return DSN does not itself request delivery notification and has an empty return path (as required for DSN).
- Do NOT use the TIMELY option on any DSN return, so that notification delivery does not fail if a non-DELIVERBY or non-TIMELY MTA is encountered on the return path.
- Use the DELIVERBY option to request timely delivery for any DSN return, using a delivery interval 2 times the original forward path DELIVERBY time (taken from the received TIMELY parameter), specifying a 'by-mode' value of 'R'. (The lack of a return path on the DSN response will mean that neither success or failure notification will be generated: if the DSN cannot be returned within the time given, it is silently dropped.)
- The sender MAY assume that the message is lost after 3 times the

original DELIVERBY interval has passed without notification.

NOTE: The above timings are based on a working assumption that normal transit times do not vary by more than a factor of two. There is nothing scientific about this choice of value, but laying down an assumption provides a basis for defining some operational parameters used by cooperating parties, which in turn provides some basis for deterministic behaviour.

The purpose of this specification is to give the sender control over the recovery strategy to be used if timely delivery does not succeed. It is therefore beyond its scope to set out exactly what recovery action the sender should take. One possible action is to retry the transmission, in which case the following additional considerations apply:

- Retries should be used very sparingly, as the likely cause of failure is either a permanent network condition or network congestion. In the case of congestion, retries are likely to make things worse. (The design of the TCP protocol takes account of many lessons about network behaviour that have been learned over the years. A particularly important strategy used is exponential back-off when retransmitting.)

- The sender is required to provide envelope ID with message. If it re-tries, it MUST use same envelope ID and SHOULD do so within a reasonable period of determining the original message has not been delivered.

- The receiver of a TIMELY message SHOULD keep note of the received envelope ID for some period, for the purpose of weeding out duplicates.

4. Timely extension to ESMTP Deliver By extension

The purpose of this extension is to allow a message sender to require that timely delivery semantics, described in this memo, be supported all along the path from message sender to receiving agent, in addition to the existing semantics of DELIVERBY.

4.1 Framework for TIMELY extension to DELIVERBY

This extends the framework template for DELIVERBY, given in RFC 2852 [5]:

(1) ESMTP extension name:
"Deliver by", extended for "timely completion"

(2) EHLO keyword:
DELIVERBY, extended as described below

(3) EHLO keyword parameters:
TIMELY (see 4.2 below)

(4) SMTP command parameters:
MAIL FROM: TIMELY (see 4.3 below)

(5) The maximum length of a MAIL FROM command line is increased by a further 17 characters for the TIMELY parameter (this being in addition to the 17 character extension for the basic DELIVERBY extension.

(6) Additional SMTP commands:
(none)

4.2 Extension to EHLO DELIVERBY keyword

This specification defines an extension token for timely completion. The extension token syntax (from RFC 2852 [5]) is extended thus:

```
extension-token /= "TIMELY"
```

An ESMTP server that supports this timely completion extension MUST also support the delivery status notification (DSN) ESMTP extension.

Support for the timely completion extension indicates support for the MAIL FROM: TIMELY parameter, described below, and for all the associated processing semantics.

4.3 MAIL FROM: TIMELY parameter

The MAIL FROM command TIMELY parameter MUST be used in conjunction with a BY parameter. Its use imposes requirements on the receiving server's handling of the message that are in addition to those

imposed by the BY parameter.

TIMELY parameter syntax:

```
timely-parameter = "TIMELY=" interval
interval         = 1*9DIGIT
```

The 'interval' is specified by the original sender of a message to be the same as the corresponding BY parameter value.

A mail relay copies the received TIMELY value to the retransmitted message, unchanged. In this way, the originally specified delivery interval is available to all MTAs that handle the message. The TIMELY value is used when generating a DSN response.

The effect of a TIMELY parameter is to require that message processing be performed in accordance with the timely completion mechanisms described in section 3 above.

5. DSN reporting extensions

This specification defines some DSN reporting extensions to allow additional status information to be returned, which a sending system might use in choosing a recovery strategy.

5.1 New extended mail system status codes

This specification uses the following additional enhanced mail system status codes, extending the range of those defined by RFC 1893 [13]:

5.4.9: Protocol required for timely delivery not supported. (Defined by [15].)

See section 3.4 for a more detailed description.

5.2 'Retry-count' per-recipient DSN header

This memo defines an additional per-recipient DSN report field 'Retry-count':

```
retry-count-field = "Retry-Count" ":" 1*3DIGIT
```

This field is used in conjunction with status code 4.4.1 to indicate the number of retries attempted before delivery was abandoned. A value of "0" means that no retries were attempted. The purpose of this is to provide information to the sender that can be used in deciding a recovery strategy.

NOTE: It is in the nature of timely completion that retries, if performed, need to be more closely spaced than is typical for SMTP retries; thus it may be necessary to reduce the number of retries to avoid overloading a relay. Some relays may choose to not attempt any retries for messages with the TIMELY option. In such circumstances, a sender may wish to retry before attempting transmission by alternative means.

6. Implementation notes

This section is not a normative part of this specification.

The timely completion mechanism is a response to requests for improved performance in certain uses of email.

Ultimately, achieving the desired performance levels is dependent on quality of implementation and operational deployment factors. If a system capable of handling 1000 messages-per-hour is subjected to periods of demand for 2000 messages-per-hour throughput, then the performance goals are bound to be substantially under-achieved, whatever the protocol specification may demand.

The rest of this section discusses some of the implementation issues and choices raised by this memo, and indicates some ways in which the performance goals can be addressed.

6.1 Message state management

All requirements for extended-term state retention are in the sending and receiving MTAs -- at or close to the edge of the network. Ideally, these would be the only MTAs involved, so provisioning of the service would be entirely under the control of the organizations that use (or sell) it.

Where intermediate relays are used, there is no requirement to maintain information about a message after it has been relayed. Thus there are no scalability problems created by a need for state maintenance; performance comes down to message throughput. The requirement for "reasonable effort" rather than "best effort" delivery for TIMELY messages means that some message handling requirements can be relaxed. Rather than copying message data to disk for re-transmission, it can be held in memory -- it might even be streamed through to the next relay; loss of message data is not critical because reporting failure back to the sender is an allowed option.

When a TIMELY MTA is subjected to high load factors, it needs a strategy for dealing with this.

The design for timely confirmation to the sender depends on reasonably consistent transit times on the forward and return message paths. Delays on the forward path are picked up and responses can be generated. Delays on the return path will result in loss of confirmation; losing failure responses should not be too damaging as the sender will time out and invoke a recovery strategy. Losing success responses is more harmful, as it may cause unnecessary additional network traffic.

In view of the above, the following message handling strategy is suggested:

- Give top priority to forwarding timely status notifications; i.e. messages with a BY parameter and no return path address.
- Give next priority to receiving new messages.
- Give next priority to processing accepted messages using the TIMELY option.
- Give next priority to forwarding messages using the DELIVERBY option.
- Finally, forward ordinary messages

If confirmation for a message sent using the TIMELY option is not received within the expected interval, the sender should be very conservative about simply retrying. The reason for non-receipt of confirmation is probably:

(a) Because of mail system congestion, in which case retransmission will just make things worse, or

(b) Some other network problem, in which case a retry won't help.

Since the motivation for this specification is to provide message delivery while the sender waits, a reasonable approach would be to give the sender an option to retry later, send by regular email or use some other delivery mechanism.

6.2 Retransmission timing issues

Even allowing for the caution stated above about the problems of simply retransmitting a failed message, it may be that some limited retransmission by the original sender is appropriate as part of a recovery strategy.

NOTE: this section draws on some well-known TCP strategies, but the primary intent is different. TCP specifies a retransmission strategy to achieve reliability. This specification aims for deterministic behaviour as far as the sender is concerned, and limits on retransmission to reduce congestion and duplicate delivery.

In order not to exacerbate congestion of intermediate relays, the following approach is suggested:

- A first retry should not be attempted before 4x the original DELIVERBY interval has expired.
- Subsequent retry attempts should be attempted at exponentially increasing intervals; e.g. 8x original interval for the 2nd retry, 16x for the 3rd retry, etc.
- The requested delivery interval should be increased exponentially for each retry.
- The total number of retries attempted should be kept reasonably small; e.g. a maximum of 3-4 retries. If a timely delivery is not achieved within a few attempts, it is probably not achievable at all within a reasonable time.
- The receiver of a message should keep a record of the received message identifier for some period of time, at least 8 times the original DELIVERBY interval, for the purpose of weeding out duplicates. It is not possible to state an absolute upper bound on this period, but it should be as long as the receiver can reasonably manage, but probably no more than a few days.

More specific recommendations for retransmission strategies may emerge from deployment experience with this protocol. The basic approach outlined above uses lessons learned from TCP, notably that exponential back-off is important to avoid exacerbating congestion conditions that may be the reason for failure in the first place.

6.3 Delivery timing granularity

This specification uses seconds for its time interval values. The best possible timing resolution for each relay is a whole number of seconds. Careless handling of these time intervals could lead to timing errors of a second or worse at each relay.

In general, it is expected that delivery time intervals will be of the order of 10s of seconds, not less than 10 seconds. The effects

of cumulative timing errors should not be significant if the number

of MTAs involved is kept small (e.g. no more than 2 intermediate relays).

The following procedure is suggested for dealing with timing through relay MTAs:

- On receipt of a MAIL FROM command, note the time at which it was received, preferably with sub-second granularity.

- When the message is subsequently forwarded, note the time immediately prior to generating the new MAIL FROM command, and use the difference from time of receipt to calculate the transit delay. The calculated transit delay should be rounded up to a whole number of seconds.

- Generate a new MAIL FROM command with the BY parameter 'by-time' value decreased by the transit delay value.

Rounding up the transit delay should mean that the BY interval is always decreased by at least 1 when passing through a relay. This should mean that if many relays are involved, the overall timing becomes more conservative. This is consistent with the idea that responsiveness to the sender is considered more important than actually achieving delivery.

6.4 Partial success

Messages sent to more than one recipient using the TIMELY option may succeed or fail independently.

Systems must be designed to handle this possibility. E.g. a sending agent that gives the user an option to resend, or send by another route, should be capable of recognizing (and reporting) that some messages have been transferred successfully, and only attempt an alternative transfer for those that did not (unless, of course, the user directs otherwise).

6.5 Routing TIMELY and non-TIMELY messages

The use of MX mail routing means that TIMELY and non-TIMELY messages to the same domain will be routed via the same servers. It may be desirable to use separate servers for TIMELY messages. One way to achieve this operationally would be to use a different email domain for TIMELY messages, but this may not be ideal from the users' view of the service.

6.6 Expediting message handling

Invoking the DELIVERBY extension for a given message may be used by an MTA as a signal to expedite message delivery. But note that status reports are part of the timely completion cycle, and while these are sent using the DELIVERBY extension, they do not use the TIMELY option. Unlike forward-path delays, any delays on the return path may directly result in the silent loss of a message status report.

This means that return path messages should be processed at least as expeditiously as the original message. Hence messages sent using the TIMELY option should not be given a higher priority than messages that use just the DELIVERBY option.

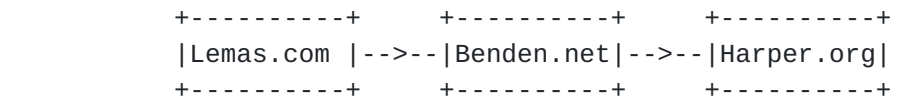
7. Examples

In the following examples, 'C:' prefixes commands sent from the SMTP client to the server in a mail transaction, and 'S:' prefixes responses from the server back to the client.

The notation '\' at the end of a command example indicates that it continues on the next line. The actual SMTP command must be presented on a single line.

7.1 Timely delivery and confirmation

This example is of a successful timely delivery and confirmation.



First hop transfer:

```
S: 220 Benden.net ESMTP server
C: EHLO Lemas.com
S: 250-Benden.net
S: 250-DELIVERBY 10,TIMELY
S: 250 DSN
C: MAIL FROM:<Asgenar@Lemas.com> BY=20;R TIMELY=20 \
   ENVID=EE271828 RET=HDRS
S: 250 OK to attempt delivery within 20 seconds
C: RCPT TO:<Robinton@Harper.org> NOTIFY=SUCCESS,FALURE \
   ORCPT=rfc822;Robinton@Harper.org
S: 250 OK
C: DATA
```

```
     S: 354 Send data
```

```
   C: (message data goes here)
       :
      .
   S: 250 Message received
```

At this point, the receiving server Benden.net has accepted responsibility to deliver the message to its destination or send a failure report back to the sender. Assuming that the next hop is initiated after a delay of 4 seconds, it may look like this:

```
   S: 220 Harper.org ESMTP server
   C: EHLO Benden.net
   S: 250-Harper.org
   S: 250-DELIVERBY 10,TIMELY
   S: 250 DSN
   C: MAIL FROM:<Asgenar@Lemas.com> BY=16;R TIMELY=20 \
      ENVID=EE271828 RET=HDRS
   S: 250 OK
   C: RCPT TO:<Robinton@Harper.org> NOTIFY=SUCCESS,FALURE \
      ORCPT=rfc822;Robinton@Harper.org
   S: 250 OK
   C: DATA
   S: 354 Send data
   C: (message data goes here)
       :
      .
   S: 250 Message received
```

At this point, the delivery MTA Harper.org has accepted responsibility to achieve message delivery and report success or to report a failure within 16 seconds of receiving the MAIL FROM command. This will depend on some kind of cooperation with the receiving user agent. When delivery is completed within the specified interval, a DSN report is sent in the following fashion:

```
   S: 220 Benden.net ESMTP server
   C: EHLO Harper.org
   S: 250-Benden.net
   S: 250-DELIVERBY 10,TIMELY
   S: 250 DSN
   C: MAIL FROM:<> BY=40;R
   S: 250 OK
   C: RCPT TO:<Asgenar@Lemas.com> NOTIFY=NEVER
   S: 250 OK
   C: DATA
   S: 354 Send data
```

```
C: To: Asgenar@Lemas.com
   From: Message-handling@Harper.org
   Subject: Disposition OK for Robinton@Harper.org
   Content-type: multipart/report; boundary=next;
                 report-type=delivery-status
   MIME-version: 1.0

   --next
   Content-type: text/plain; charset=US-ASCII

   Your message (EE271828) to <Robinton@Harper.org> was processed
   --next
   Content-type: message/delivery-status

   reporting-MTA: dns; mail-receiver.Harper.org
   Original-Envelope-ID: EE271828

   Original-recipient: rfc822;Robinton@Harper.org
   Final-recipient: rfc822;Robinton@Harper.org
   Action: delivered
   Status: 2.0.0
   --next--
   .
S: 250 Message received
```

On receipt of this confirmation message, the sender's user agent will be able to correlate with the original using the 'Original-Envelope-ID' and 'Original-recipient' values, and confirm to the sender that the message has been delivered and processed.

7.2 Received by delivery MTA and timed out

This example follows the same sequence as the previous one, up to the point that the delivery MTA Harper.org has accepted responsibility to achieve message delivery or to report a failure. In this case, having accepted the message, final delivery cannot be achieved in the desired interval so a failure DSN must be sent:

```
S: 220 Benden.net ESMTP server
C: EHLO Harper.org
S: 250-Benden.net
S: 250-DELIVERBY 10,TIMELY
S: 250 DSN
C: MAIL FROM:<> BY=40;R
S: 250 OK
C: RCPT TO:<Asgenar@Lemas.com> NOTIFY=NEVER
S: 250 OK
C: DATA
```

```
     S: 354 Send data
```

```
C: To: Asgenar@Lemas.com
   From: Message-handling@Harper.org
   Subject: Disposition failed for Robinton@Harper.org
   Content-type: multipart/report; boundary=next;
                 report-type=delivery-status
   MIME-version: 1.0

   --next
   Content-type: text/plain; charset=US-ASCII

   Your message (EE271828) to <Robinton@Harper.org> could not
   be processed within the requested time.
   --next
   Content-type: message/delivery-status

   reporting-MTA: dns; mail-receiver.Harper.org
   Original-Envelope-ID: EE271828

   Original-recipient: rfc822;Robinton@Harper.org
   Final-recipient: rfc822;Robinton@Harper.org
   Action: failed
   Status: 4.4.7 (Timed out during delivery)
   --next--
   .
S: 250 Message received
```

Because this is a specific failure condition being sent to a source that has used the timely delivery extension, and the message can be correlated with the original by means of the 'Original-Envelope-ID' and 'Original-Recipient' values, no part of the original message is returned with the DSN report.

7.3 Timed out with delivery in progress

This example is similar to the previous one, except that final delivery is in progress but its completion is delayed. In this case, the message cannot be recalled, so a notification report is sent to the sender, within the requested delivery period, indicating that the message is delivered and in delivery. Later, a final delivery status message will be sent.

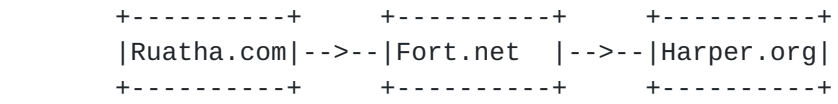
```
   S: 220 Benden.net ESMTP server
   C: EHLO Harper.org
   S: 250-Benden.net
   S: 250-DELIVERBY 10,TIMELY
   S: 250 DSN
   C: MAIL FROM:<> BY=40;R
   S: 250 OK
   C: RCPT TO:<Asgenar@Lemas.com> NOTIFY=NEVER
   S: 250 OK
   C: DATA
   S: 354 Send data
   C: To: Asgenar@Lemas.com
      From: Message-handling@Harper.org
      Subject: Disposition delayed for Robinton@Harper.org
      Content-type: multipart/report; boundary=next;
                    report-type=delivery-status
      MIME-version: 1.0

      --next
      Content-type: text/plain; charset=US-ASCII

      Your message (EE271828) to <Robinton@Harper.org> is being
      delivered but not completed within the requested time.
      --next
      Content-type: message/delivery-status

      reporting-MTA: dns; mail-receiver.Harper.org
      Original-Envelope-ID: EE271828

      Original-recipient: rfc822;Robinton@Harper.org
      Final-recipient: rfc822;Robinton@Harper.org
      Action: delayed
      Status: 4.4.7 (Timed out during delivery)
      --next--
      .
   S: 250 Message received
```

The difference between this and the example in section 7.2 is in

the "Action:" field of the delivery status message.

7.4 Timed out before receipt by delivery MTA

This example is of a failed attempt to achieve timely delivery because the message could not be forwarded within the requested interval.



First hop transfer:

```
S: 220 Fort.net ESMTP server
C: EHLO Ruatha.com
S: 250-Fort.net
S: 250-DELIVERBY 15,TIMELY
S: 250 DSN
C: MAIL FROM:<Jaxom@Ruatha.com> BY=20;R TIMELY=20 \
   ENVID=EE271828 RET=HDRS
S: 250 OK to attempt delivery within 20 seconds
C: RCPT TO:<Sebell@Harper.org> NOTIFY=SUCCESS,FALURE \
   ORCPT=rfc822;Sebell@Harper.org
S: 250 OK
C: DATA
S: 354 Send data
C: (message data goes here)
    :
   .
S: 250 Message received
```

After a delay of 12 seconds (with 8 seconds of the original delivery interval remaining), the server Fort.net attempts to relay the message:

```
S: 220 Harper.org ESMTP server
C: EHLO Fort.net
S: 250-Harper.org
S: 250-DELIVERBY 10,TIMELY
S: 250 DSN
C: QUIT
S: 221 <Harper.org> closing channel
```

The minimum delivery interval declared by the server Harper.org is greater than the time remaining to complete delivery, so Fort.net does not even attempt to send the message. Instead, it returns a failure report back to Ruatha.com:

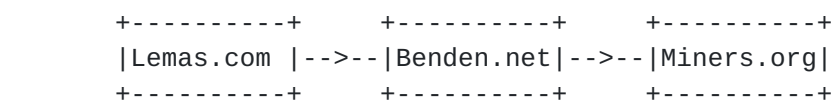
```
   S: 220 Ruatha.com ESMTP server
   C: EHLO Fort.net
   S: 250-Ruatha.com
   S: 250-DELIVERBY 10,TIMELY
   S: 250 DSN
   C: MAIL FROM:<> BY=40;R
   S: 250 OK
   C: RCPT TO:<Jaxom@Ruatha.com> NOTIFY=NEVER
   S: 250 OK
   C: DATA
   S: 354 Send data
   C: To: Jaxom@Ruatha.com
      From: Message-handling@Fort.net
      Subject: Delivery failed for Sebell@Harper.org
      Content-type: multipart/report; boundary=next;
                    report-type=delivery-status
      MIME-version: 1.0

      --next
      Content-type: text/plain; charset=US-ASCII

      Your message (EE271828) to <Sebell@Harper.org> could not be
      delivered within the requested time.
      --next
      Content-type: message/delivery-status

      reporting-MTA: dns; mail-relay.Fort.net
      Original-Envelope-ID: EE271828

      Original-recipient: rfc822;Sebell@Harper.org
      Final-recipient: rfc822;Sebell@Harper.org
      Action: failed
      Status: 4.4.7 (Timed out during message transfer)
      Retry-count: 0
      --next--
      .
   S: 250 Message received
```

From the sender's perspective, this is pretty much the same condition as reported by example 7.2, the difference being that the time-out has occurred before the message reaches the delivery MTA. The status code is the same but the reporting MTA is different.

The retry count value is returned to give the sender an indication about whether it might retry this path before switching to an alternative delivery strategy.

7.5 Timely delivery feature not supported

This final example shows failure of a timely delivery request because a receiving MTA does not support the capability:



First hop transfer:

```
S: 220 Benden.net ESMTP server
C: EHLO Lemas.com
S: 250-Benden.net
S: 250-DELIVERBY 10,TIMELY
S: 250 DSN
C: MAIL FROM:<Asgenar@Lemas.com> BY=20;R TIMELY=20 \
   ENVID=EE271828 RET=HDRS
S: 250 OK to attempt delivery within 20 seconds
C: RCPT TO:<Nicat@Miners.org> NOTIFY=SUCCESS,FALURE \
   ORCPT=rfc822;Nicat@Miners.org
S: 250 OK
C: DATA
S: 354 Send data
C: (message data goes here)
    :
   .
S: 250 Message received
```

Five seconds later, Benden.net attempts to forward the message:

```
S: 220 Miners.org ESMTP server
C: EHLO Benden.net
S: 250-Harper.org
S: 250-DELIVERBY 60
S: 250 DSN
C: QUIT
S: 221 <Miners.org> closing channel
```

The Miners.org server does not support timely delivery, so Benden.net does not attempt to send the message. Instead, it sends a failure report back to Lemas.com:

```
   S: 220 Lemas.com ESMTP server
   C: EHLO Benden.net
   S: 250-Lemas.com
   S: 250-DELIVERBY 10,TIMELY
   S: 250 DSN
   C: MAIL FROM:<> BY=40;R
   S: 250 OK
   C: RCPT TO:<Asgenar@Lemas.com> NOTIFY=NEVER
   S: 250 OK
   C: DATA
   S: 354 Send data
   C: To: Asgenar@Lemas.com
      From: Message-handling@Benden.net
      Subject: Delivery failed for Nicat@Miners.org
      Content-type: multipart/report; boundary=next;
                    report-type=delivery-status
      MIME-version: 1.0

      --next
      Content-type: text/plain; charset=US-ASCII

      Your message (EE271828) to <Nicat@Miners.org> could not be
      delivered within the requested time.
      --next
      Content-type: message/delivery-status

      reporting-MTA: dns; mail-relay.Benden.net
      Original-Envelope-ID: EE271828

      Original-recipient: rfc822;Nicat@Miners.org
      Final-recipient: rfc822;Nicat@Miners.org
      Action: failed
      Status: 5.4.9 (Timely delivery not supported by Miners.org)
      --next--
      .
   S: 250 Message received
```

8. IANA Considerations

This specification introduces some new protocol elements for which IANA registration be required or desirable:

- Extension to DELIVERBY ESMTP extension: see section 4.
- New extended mail system status codes: see section 5.1.
- New DSN report per-recipient field: see section 5.2.

9. Internationalization considerations

This specification introduces no new internationalization considerations other than those already present in DSN, which, through MIME, provides for charset identification and language tagging of the human readable part of a DSN report.

10. Security considerations

See also RFC 1894 [12], RFC 2852 [5].

To offer timely handling of messages may require some dedication of resource. It is conceivable that systems supporting this feature may be more susceptible to denial of service attacks from a flood of messages requesting timely completion. (See also section 6.1.)

There is a distant possibility that responses to time-sensitive requests may disclose information about the loading or topology of the network accessed. This is unlikely to be any worse than for web access protocols (but note that HTTP has been shown to allow certain kinds of timing attack on private information about a client's network activities.).

Systems that depend on the physical presence of a user to achieve timely receipt SHOULD NOT accept a message for such disposition without the user's explicit permission (c.f. automated generation of MDN responses in RFC 2998 [14]).

11. Acknowledgements

The authors thank Hiroshi Tamura-san for undertaking the task of reviewing a very rough, early draft and making several pertinent observations. The authors also acknowledge helpful comments by Dan Wing, Ned Freed and Greg Vaudreuil.

12. References

[1] RFC 2542, "Terminology and Goals for Internet Fax"
L. Masinter, Xerox Corporation
March 1999.

[2] RFC 821, "Simple Mail Transfer Protocol"
Jonathan B. Postel, ISI/USC
August 1982.

[3] RFC 1651, "SMTP Service Extensions"
J. Klensin, MCI
N. Freed, Innosoft
M. Rose, Dover Beach Consulting, Inc.
E. Stefferud, Network Management Associates, Inc.
D. Crocker, Silicon Graphics, Inc.
July 1994.

[4] RFC 1891, "SMTP Service Extension for Delivery Status Notifications"
K. Moore, University of Tennessee
January 1996.

[5] RFC 2852, "Deliver By SMTP Service Extension"
D. Newman, Sun Microsystems
June 2000

[6] "Content Negotiation for Internet Messaging Services"
G. Klyne, Baltimore Technologies
R. Iwazaki, Toshiba TEC
D. Crocker, Brandenburg Consulting
Internet draft: draft-ietf-fax-content-negotiation-05.txt
Work in progress: May 2001.

[7] RFC 2234, "Augmented BNF for Syntax Specifications: ABNF"
D. Crocker (editor), Internet Mail Consortium
P. Overell, Demon Internet Ltd.
November 1997.

[8] "Procedures for document facsimile transmission in the general switched telephone network"
ITU-T Recommendation T.30 (1996), including Amendment 1 (1997), Amendment 2 (1997), Amendment 3 (1998) and Amendment 4 (1999)
International Telecommunications Union.

[9] RFC 1047, "Duplicate messages and SMTP"
C. Partridge
February 1988.

[10] RFC 974, "Mail routing and the domain system"
C. Partridge
January 1986.

[11] RFC 2119, "Key words for use in RFCs to Indicate Requirement Levels"
S. Bradner, Harvard University
March 1997.

[12] RFC 1894, "An Extensible Format for Delivery Status Notifications"
K. Moore, University of Tennessee
G. Vaudreuil, Octel Network Services
January 1996.

[13] RFC 1893, "Enhanced Mail System Status Codes"
G. Vaudreuil, Octel Network Services
January 1996.

[14] RFC 2298, "An Extensible Message Format for Message Disposition Notifications"
R. Fajman, National Institutes of Health
March 1998.

[15] G. Vaudreuil, Lucent Technologies,
Extensions to Mail System Status Codes
Internet draft: draft-vaudreuil-1983ext-01.txt
Work-in-progress, November 2001.

13. Authors' addresses

Graham Klyne (editor)
MIMEsweeper Group,
1310 Waterside,
Arlington Business Park
Theale
Reading, RG7 4SA
United Kingdom.
Telephone: +44 118 903 8000
Facsimile: +44 118 903 9000
Email: Graham.Klyne@MIMEsweeper.com

David H. Crocker
Brandenburg Consulting
675 Spruce Drive
Sunnyvale
CA 94086
USA.
Telephone: +1 408 246 8253
Facsimile: +1 408 273 6464
Email: dcrocker@brandenburg.com

Appendix A: Amendment history

[[[RFC editor: please remove this appendix on publication]]]

00a	22-Oct-1999	Memo initially created.
01a	13-Sep-1999	Incorporate review comments. Update references. Changed title. Incorporate material from IETF meeting presentations.
02a	25-Jan-2001	Update author details. Simplify COMPLIANCE extension to a TIMELY extension of DELIVERBY. Add original interval parameter to TIMELY option. Strengthen description of mechanism for timely confirmation. Add template decsription for TIMELY extension. Refer to the goal of this specification as "timely completion" rather than just "timely delivery" (to clearly distinguish from basic DELIVERBY). Added subsection dealing with final MTA/MUA interaction. Defined DSN extension header and status codes for reporting timely delivery failures. Drafted some implementation notes.
02b	30-Jan-2001	Add examples. Update some references. Other editorial drafting.
02c	31-Jan-2001	Fold in review comments. Added implementation note about using DELIVERBY to expedite message handling (6.5).
02d	01-Feb-2001	More editorial changes.
02e	01-Feb-2001	Revised text dealing with time-out; move discussion of retries to implementation notes.
03a	16-Feb-2001	Editorial changes. Added some clarifying text to

introductory section 2.2.

03b	13-Jun-2001	Editorial changes.
04a	12-Sep-2001	Rework some descriptions and status codes to be clear that this document describes an extension of the mail delivery semantics rather than some aspect of disposition semantics. As a result, some status codes have been removed. Removed text in section 6.1 about immediate rejection of a message to help avoid exacerbating congestion conditions. Changed terminology to focus on the goal of achieving "final receipt" (rather than "disposition"). Added reference [15].
04b	13-Sep-2001	Change contact details. Editorial corrections and refinements.
04c	13-Sep-2001	Changed some section titles; revised examples commentary and status codes in line with other changes.
05a	17-Sep-2001	Reworked abstact and introduction to more clearly place this work in context. Various minor editorial changes.
06a	05-Nov-2001	Prepare for WG last call. Fixed extended status code for "protocol not supported". Removed editorial notes. Fix some spelling errors.

REVIEW CHECKLIST:

(Points to be checked or considered more widely on or before final review.)

- Are there any deployed mechanisms that MTAs may use to recognize expedited message relay?

- Possible minor revision to DELIVERBY spec? If a DELIVERBY MTA fails message delivery because the delivery time has expired, AND the message has an empty SMTP sender address/return path, the message should be silently discarded (c.f. RFC 1891, section 6.2; I think the considerations noted there seem less applicable.). If this doesn't work, try next...

- Consider addition of new 'by-mode' value for return DSNs; e.g. 'E' for expedite: try to deliver within interval given, or abandon delivery, but don't notify success or failure. (Currently specify 'R' without return-path.) A notification should not be abandoned if a non-DELIVERBY MTA is encountered.

- Try to model system behaviour under high-load/backlog conditions. Especially w.r.t. section 3.5.

- What lessons to learn from IP QoS efforts?

- Query use of enhanced status codes 4.x.x and 5.x.x; Use by DELIVERBY seems at odds with RFC 1893.

- Note use of status 5.4.1 not in line with expectations of RFC 1893.

- Use new code instead of 5.3.3?

- Special considerations for fax offramp gateay? How to deal with uncertain dial-out times.

- Apparently, DSN extension fields must be registered with IANA, but there appears to be no registry for them.

- Use of alternative port? (e.g. like message submission).

- Allow MTAs to impose size limit on messages for timely delivery?

- Operational issues surrounding selection of delivery interval?

- DISCUSS: In environments where the timing of final delivery of the message is outside the control of the final MTA (e.g. the time required for an outdial, or waiting for a client to collect the message), an interim DSN report may be generated indicating that the message has been received pending final delivery. This report should be clear whether final delivery is dependent on the receiving user (e.g. mail collection) or some other unknown infrastructure delay (e.g. fax out-dial or external e-mail environment).

  This is covered somewhat by section 3.3.1: is this adequate?

- MX configuration -- uniform routing for TIMELY/non-TIMELY. Is a differential routing option required; e.g. SRV records?

- Can use of ORCPT be relaxed? If partial success occurs for multiple recipients, it is important to be able to tell which were successful and which were not.

- When a timely-delivery failure message is sent back, it is addressed to the sender of the original message; thus it becomes the sender UA responsibility to handle the failure of timely delivery -- does this cause any problems?

- Check examples. (Should relays declare mail-domain or host name? Does it matter? Should the From: header for DSNs always be 'postmaster', or is any appropriate mailbox OK?)

Full copyright statement