Approved for Public Release; Distribution Unlimited
Case # 06-1000

Custodial Multicast in Delay Tolerant Networks

Challenges and Approaches

Susan Symington, Robert C. Durst, and Keith Scott
The MITRE Corporation
McLean, Virginia
susan@mitre.org, durst@mitre.org, kscott@mitre.org

***Abstract*— Although custodial transmission of multicast bundles would be a desirable capability to have in Delay-Tolerant Networks (DTNs), support for custodial multicast transmission was omitted from the Bundle Protocol Specification because of its complexity. This paper explains the difficulties that arise with respect to supporting custody transfer and retransmission of multicast bundles, and it describes some potential solutions for addressing these issues that the authors are currently exploring as work-in-progress.**

Keywords-Delay-Tolerant, Disruption-Tolerant, multicast, custody transfer, custody-based retransmission

I. INTRODUCTION AND BACKGROUND

The success of the Internet is due, in part, to the fact that it consists primarily of relatively stable links. If something occurs to disrupt the continuous connectivity that we have come to rely on with the Internet, however, the performance of its protocols, if they work at all, is severely degraded.

Delay Tolerant Networking (DTN) as a research area is focused on addressing the communication requirements specific to challenged networks, i.e., networks that may suffer frequent, possibly unpredictable disconnection, high delay, high data rates, or asymmetric data rates between source and destination. The DTN Network Architecture [1] and Bundle Protocol Specification [2] that have been developed within the DTN Research Group of the Internet Research Task Force have emerged as feasible solutions for improving network performance in challenged environments (as well as for working well in stable environments). DTN uses a store-and-forward message overlay system that spans regional subnetworks. Instead of relying on chatty, end-to-end protocols that require sources and destinations to exchange many packets in the course of a given communication, DTN aggregates data into bundles. It minimizes the need for end-to-end connectivity and is able to store bundles in persistent storage at various points as they travel through the network so that if connectivity is lost, the bundles need only be retransmitted from their closest storage point rather than all the way from the source.

The basic message delivery service provided by the Bundle Protocol is unacknowledged and is not guaranteed. A key Bundle Protocol innovation designed to enhance delivery reliability is known as custodial delivery. The bundle protocol specifies the procedures for supporting custodial delivery of bundles that are destined for a single destination node. However, it does not discuss how custodial delivery could be provided for a bundle that is destined for multiple destination nodes, i.e., a multicast bundle. Nevertheless, there is a strong motivation for wanting to be able to use custodial multicast in challenged networks because of its potential to conserve the already-scarce resource of bandwidth by both streamlining the transmission path and minimizing the resources used by retransmissions. This document describes some of the issues unique to providing support for custodial multicast and it describes some potential solutions for addressing these issues that the authors are currently exploring as work-in-progress [3].

This work was supported by DARPA *under contract #DAAB07-03-C-N206 and is subject to the "Rights in Data" clause at DFARS 252.227-7013 (NOV 95). This work was performed at The MITRE Corporation.*

II. CUSTODIAL UNICAST VERSUS CUSTODIAL MULTICAST

As described in the Bundle Protocol, each bundle is sent to an endpoint, which may be a set of zero or more nodes. If an endpoint never contains more than one node, it is called a "singleton" endpoint. A bundle that is sent to a singleton endpoint is what is conventionally thought of as a bundle that is unicast. If an endpoint may contain more than one node such that a bundle that is sent to that endpoint is expected to reach all of the nodes in that endpoint, then such a bundle is what is conventionally thought of as a bundle that is multicast. The Bundle Protocol describes the mechanisms required to support custodial transfer of singleton bundles, but it makes clear that those custodial delivery mechanisms are not applicable to bundles that are sent to non-singleton, e.g. multicast, endpoints.

A. How Singleton Custodial Transfer Works

As currently defined in the Bundle Protocol, custody transfer is supported as follows: any bundle may have its "Custody transfer is requested" flag set. When a node receives a bundle with this flag set, that node may:

- Take custody of the bundle and send a custody transfer success signal to the previous custodian,
- Forward the bundle without taking custody, or
- Send a custody transfer failure signal to the previous custodian indicating that the bundle is being deleted (which may happen, for example, if the bundle can neither be forwarded nor stored by the receiving node).

A custodian of a bundle should store that bundle—in persistent storage if possible—until either the bundle is delivered at its destination or custody of it is transferred to another node. A custodian should have a retransmission timer associated with

Approved for Public Release; Distribution Unlimited
Case Number 06-1000
©2006-The MITRE Corporation. All rights reserved.

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2006		**00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Custodial Multicast in Delay Tolerant Networks	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **The MITRE Corporation,7515 Colshire Drive,McLean,VA,22102-7539**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**5**	

the bundle and may retransmit the bundle in response to this timer timing out or in response to a failed custody signal (as appropriate, depending on the nature of the failure). Other than the issue of selecting an appropriate value for the retransmission timer for a given bundle, custody transfer of singleton bundles is relatively straightforward.

B. How Multicast Complicates Custodial Delivery

Custody transfer and custody-based retransmission are fundamentally complicated when applied to a multicast bundle because any node in the delivery path of a multicast bundle may be a branching node and branch the bundle (i.e., copy the bundle and both deliver and forward it, or copy the bundle and forward it to multiple next-hop nodes). Neither the branching nodes themselves nor the number of copies of a given bundle that will be created at any given branching node is necessarily known by the current custodian of the bundle. Therefore, to support custodial delivery of multicast bundles, mechanisms must be defined to enable a custodian of one of these bundles to determine when all downstream copies of the bundle have either been delivered or have been taken custody of. In addition, for purposes of conserving bandwidth, it would be desirable to define mechanisms to enable bundles to be re-forwarded selectively, to only those downstream branches of the delivery path that have not yet received them, rather than indiscriminately, to all downstream nodes.

III. The Objectives of Custodial Delivery

A. Objective 1: Increase the likelihood that the bundle will be delivered to its destination endpoint before expiration.

The main objective of custodial delivery is to increase the likelihood that a bundle that is sent will in fact be delivered to its destination endpoint. In the multicast case, in particular, this means increasing the likelihood that a bundle will be delivered to as many of the nodes in its destination endpoint as possible.

B. Objectve 2: Reduce the Cost of Bundle Retransmissions

A second objective of custodial delivery is to increase the likelihood that if a bundle needs to be re-forwarded, the cost of re-forwarding it from the custodian, in terms of the routing metric in use, will be less than the cost of re-forwarding it from the source node or, ideally, from any previous custodial nodes. This objective has unique implications when applied to a multicast bundle, which may be delivered to multiple nodes, each of which may be closest to a different custodian.

The routing metric in use may be a simple one, such as topological proximity. It may be more complex, taking into consideration scheduled disconnections, link bandwidth or available persistent storage at various nodes. It should be designed to offer custody of a bundle to nodes that, although they may not be topologically closer to the bundle's destination at the present time, will render the bundle most likely to reach its destination before expiration, ideally at a lower cost than would have been possible from previous custodians.

Given that DTN networks by definition are expected to suffer from some degree of instability, any specific instance of custody transfer of a bundle from one node to another, despite intentions, may not in fact represent progress. As long as the network characteristics on which the routing metrics are based are stable enough to enable eventual delivery of the bundle, however, then as custody of a bundle is transferred from one node to another, that bundle is, in general, progressing closer to a lower-cost retransmission point. (Determining what the stability requirements might be and how long they must be maintained in order to ensure delivery suggests itself as an interesting area of research.)

C. Objective 3: Enable delivery to late-registering nodes

A third objective of custodial delivery is to enable delivery to a node whose registration request may be late or delayed such that the registration request had not yet propagated through the network sufficiently to graft the destination node onto the distribution tree when the bundle was sent.

If a bundle has a singleton destination endpoint ID (EID), custodial delivery enables it to be stored in the network until the destination node registers and the notification of this registration request is able to propagate to the custodian (providing the bundle doesn't expire first) so that the custodian can forward the bundle toward the destination. As currently defined, once such a bundle that is sent to a singleton EID reaches the (single) node that registers with that EID, the bundle may be deleted because it will not need to be delivered to any other destination node. If a bundle has a multicast destination EID, on the other hand, there is no inherent limit to the number of such registrations that may be received. Custodians of multicast bundles may store those bundles in the network until they expire in order to ensure that the bundles will be available for forwarding to every node that has a late or delayed registration. As pointed out in [4], due to the unique characteristics of frequent partitioning and large transfer delays in DTNs, destination endpoint registration changes during data transfer may be the norm rather than the exception.

IV. Assumptions and Design Principles

A. Branching points are not required to be custodians.

It is not feasible to require branching point nodes to take custody of bundles because there is no guarantee that such nodes will always have the storage capacity to do so. Our solution assumes that branching point nodes may not become custodians of the bundles that they branch.

B. Non-custodial branching nodes must maintain state for each custodially-transferred bundle that they branch.

If a node branches a bundle without becoming custodian of it, the custodian of the bundle has no way of being aware that it is now responsible for more than one copy of the bundle. If the bundle has a singleton destination EID, then this is not a problem because ensuring the custodial transfer of only one of these bundle copies is sufficient to enable the delivery of the bundle to all nodes (i.e., the sole node) in its destination endpoint. If the bundle has a multicast destination EID, however, then (assuming that the multicast distribution path is a tree) the non-custodial branching node must keep track of the delivery/custody status of each of the copies that it creates and

in turn report this custody status information to its nearest upstream custodian or non-custodial branching point.

C. *DTN multicast routing protocols must enforce certain restrictions when supporting custodial delivery*

Because nodes that do not implement the optional extensions to support custodial multicast will not be capable of maintaining the state information required of them by assumption B above, these non-custodial-multicast-capable (non-CMC) nodes must not be allowed to be branching points in custodial multicast delivery trees. In order for custodial multicast to work in a network that consists of both CMC and non-CMC nodes, the DTN multicast routing protocols must permit only CMC nodes to be branching points.

DTN multicast routing protocols are required to operate over trees rather than meshes. The purpose of this requirement is to facilitate the detection of unintentional loops. Because the delivery path is required to be a tree, any bundle that is received a second time may be assumed either to have been re-forwarded by a custodian or to be in an unintentional loop.

D. *Convergence-layer multicast may be used, with some restrictions.*

If DTN multicast is running over an underlying multicast-capable convergence layer protocol, a bundle sent to a single DTN EID that is bound to an underlying multicast group address could be multicast to its multiple next-hop nodes using that convergence layer. In order for such a configuration to be able to be used to support custodial multicast, the bundle node that is forwarding the bundle onto the convergence layer:

- must be a CMC-node, and
- must know the number of next-hop nodes that the bundle is expected to reach using that convergence layer, so it can maintain state for this number of copies

Again, the DTN multicast routing protocols are assumed to be responsible for enforcing these requirements.

E. *Bandwidth conservation is given priority over robustness of delivery by default, but local policy may override this.*

There is an inherent tradeoff between robustness of delivery and bandwidth conservation when custody transfer of multicast bundles is requested. A single bundle may have many branching points in its distribution tree. If most copies of the bundle reach their intended destination but only a few do not, it would conserve bandwidth if the bundle could be retransmitted from custodians only along those paths of the distribution tree leading to non-receivers so that those branches of the tree that lead to nodes that have already successfully received the bundle would not have to waste bandwidth transmitting and sending custody signal for the bundle multiple times.

If the network is not stable for the duration of a bundle delivery, however, then a non-delivery branch of the distribution path may no longer lead to an intended recipient node. In fact, the only path to that intended recipient may now be from some other portion of the tree that the bundle has already successfully traversed. To increase the likelihood that a multicast bundle will be delivered to all intended destination nodes in an unstable or dynamic network, therefore, a custodian that re-forwards a bundle could re-forward the bundle to all neighboring nodes that lead to destination nodes, including those neighbors that have already successfully received and forwarded the bundle.

By default, our design favors bandwidth conservation. Neither custodians nor non-custodial branching nodes should forward a multicast bundle to a next-hop node from which a successful custody signal for that bundle has already been received. Bundles are only re-forwarded to those next-hop nodes that have downstream copies of the bundle that have not yet been delivered or taken custody of. Because the multicast delivery path is required to be a tree, if a node receives a bundle of which it has already been a custodian, the bundle is assumed to be in an unintentional loop and is dropped.

V. Bundle Protocol Extensions for Supporting Custodial Multicast

The following are circumstances under which a custodian may want to re-forward a multicast bundle:

- Receipt of a "Failed" custody signal for this bundle from a specific node; assuming network stability, the custodian may want to conserve bandwidth by sending the bundle directly to this node for further forwarding
- Time-out of one or more of the custody timers for the bundle, in which case the custodian wants to re-forward the bundle (at least) on all downstream branches of the distribution path associated with the expiring timer(s)
- Notification of a new registration for a multicast EID

The protocol extensions defined in this section do not address the mechanics of re-forwarding bundles to newly-registering nodes. They do, however, enable a custodian to determine whether a bundle needs to be re-forwarded by ensuring that the custodian will be able to:

- receive "Failed" custody signals for arbitrary bundle copies from nodes downstream in the delivery path
- be aware of whether or not there exists a downstream copy of that bundle that has not been delivered or taken custody of when its custody timer times out.

Custody status notification is provided to each custodian of a multicast bundle in the same way that it is provided to each custodian of a singleton bundle: by having downstream nodes send either “Succeeded” or “Failed” custody signals to the custodian, as appropriate. In the multicast case, however, the custodian must keep track of the custody status of each copy of each bundle it forwards. When the custodian receives a “Succeeded” custody signal for each of the copies that it branched, the custodian is assured that every downstream copy has either been delivered or taken custody of. Until the custodian receives such a signal for any given copy that it forwarded, it must assume that there is at least one copy of the bundle on that copy’s branch of the distribution tree that has neither been delivered nor taken custody of.

Although the custodian expects one “Succeeded” custody signal per bundle copy that it branched, there may be more copies of the bundle created downstream of it. These copies, however, must be kept track of by the non-custodial branching nodes that create them. When all copies created by a non-custodial branching node have either been delivered or taken custody of, the branching node sends a “Succeeded” custody signal to report this to the bundle’s previous upstream custodian or branching node, and so forth, until the status of every copy that the custodian branched is reported to the custodian. There is no need for the custodian itself to be made aware of every copy of the bundle that is created downstream of it. To achieve this "relayed" custody signal transmission, every non-custodial node that is a branching point for a multicast bundle, upon receipt of that bundle, takes note of its current custodian and then places its own EID into the bundle to list itself as custodian for that bundle before forwarding the bundle (even though it really is not the custodian of the bundle in the sense that it is not storing a copy of the bundle in permanent storage, nor does it consider itself responsible for retransmitting the bundle). In this way, each branching point node is assured that it will receive any custody signals that may be generated for the bundle copies that it branches.

Non-CMC nodes are permitted to deliver custodial multicast bundles and to forward custodial multicast bundles to a single next-hop node. They are not, however, permitted to be custodians or branching points for custodial multicast bundles. Only CMC nodes may take on these roles for custodial multicast bundles.

A. Responsibilities of Custodians of Multicast Bundles

A custodian of a multicast bundle has responsibilities similar to those of a custodian of a singleton bundle. However, the custodian of a multicast bundle must maintain custody-related state information per bundle copy that it branches rather than just per bundle. Specifically, the custodian will:

- maintain a list of the copies of that bundle that it has branched along with the EID/convergence layer to which each copy was delivered or forwarded
- keep track of the "Succeeded" custody signals received
- retain the multicast bundle that it takes custody of--in persistent storage if possible-- until the bundle expires or until it receives a successful custody signal for each of the copies of the bundle that it branched
- maintain at least one retransmission timer for the bundle; possibly one timer per copy it has branched
- retransmit the corresponding copy (or copies) of the bundle upon retransmission timer expiration
- retransmit a copy of the bundle referred to by a "Failed" custody signal, perhaps encapsulated [5] in a unicast bundle sent to the “Failed” signal’s source
- destroy a retransmission timer when “Succeeded” custody signals for all bundle copies associated with that timer have been received
- delete a multicast bundle and all its associated custodial state information when the bundle expires
- maintain a list of unexpired bundles for which it has ever been a custodian

B. Responsibilities of Non-custodial Branching Nodes of Custodially-transferred Multicast Bundles

In order to act as a non-custodial branching point of a custodial multicast bundle, a node must pose as a proxy custodian by inserting its own EID into the custodian field of the bundle so that it will receive custody signals (if sent) for all copies of the bundle that it branches. In particular, it must:

- maintain a list of the copies of that bundle that it has branched along with the EID/convergence layer to which each copy was delivered or forwarded
- keep track of the "Succeeded" custody signals received
- notify the appropriate upstream node (e.g. the node that had been listed as the bundle's custodian when the bundle was received, which is either the bundle's "real” custodian or the most recent proxy custodian that may in turn pass the signal upstream) when a “Succeeded” custody signal is received for all of the copies of the bundle that it branched
- If custody transfer failure is reported for any of the downstream copies that the bundle branched, it must generate a replacement "Failed" custody signal to the appropriate upstream node and insert a Proxy EID extension block [3] into this bundle (if there is not one in there already) that identifies the source of the original "Failed" custody signal
- Upon receipt of a custodial multicast bundle, determine not only to which next-hop nodes the bundle should be forwarded, but also from which of these next-hop nodes the branching node has already received a "Succeeded" custody signal for this bundle. By default, but subject to network stability conditions and local policy, the bundle should be forwarded to only those next-hop nodes for which "Succeeded" custody signals for the bundle have not been received

As stated in the assumptions, all non-custodial branching point nodes must maintain state information for each custodially-transferred bundle that they branch, possibly until the bundle expires. Even though this state information is not expected to take up as much room as would storing the bundle itself, it is conceivable that a non-custodial branching node could find itself in a situation in which it does not have sufficient resources to maintain the state information required of it in order for it to branch a particular multicast bundle. In this case, the non-custodial branching node must send a "Failed" custody signal to the appropriate upstream node. It may also (subject to policy) reset the custody transfer requested bit and forward the bundle. If the branching node is delivering or forwarding the bundle non-custodially, the reason code in the "Failed" custody signal must be the new reason code, "Bundle Forwarded Non-Custodially". If the branching node is not delivering or forwarding the bundle non-custodially, the

reason code in the "Failed" custody signal must be "Depleted Storage". In addition, if the node's resource depletion is expected to last a while, the node may change the way it represents itself to the multicast routing protocol (subject to the specifics of that protocol), thereby actively seeking to not be a branching point in any multicast distribution paths.

C. *Identifying Bundle Copies*

The node responsibilities described above require branching nodes to be able to distinguish the various copies of a given bundle that they deliver or forward from each other, so that when a custody signal is received, the receiving node can determine not only which bundle it refers to, but which particular copy of the bundle that was branched it refers to. Furthermore, in order to perform bandwidth-efficient retransmissions that target only those next-hop nodes that still have downstream copies of the bundle for which "Succeeded" custody signals have not been received, the node receiving a custody signal must be able to determine to which next-hop node the particular copy of the bundle referred to by that signal had been forwarded. Because we cannot rely on the fact that the custody signal for a given bundle copy will always be returned via the same route along which the bundle was forwarded, the requirement that branching nodes be able to distinguish among bundle copies requires the branching node to somehow mark each bundle copy uniquely. This marking could be accomplished using a to-be-defined bundle extension header, but such a technique would require each branching node to add such a header, thereby adding to the size of the bundle. It would also require that procedures be defined for determining when these headers could be deleted from the bundle, and in general it would complicate the protocol. Instead, we recommend that a certain portion of the EID that the branching node inserts into the bundle's custodian field be used as a copy identifier. For example, suppose a node is forwarding a multicast bundle to two different next-hop nodes. This node could be registered in two singleton EIDs, e.g., *NodeA:1* and *NodeA:2* and it can be uniquely identified by only the scheme name (e.g., "NodeA:") portion the EID [6]. The node places EID *NodeA:1* in the current custodian field of the bundle that it forwards to the first next-hop node and EID *NodeA:2* in the current custodian field of the bundle that it forwards to the second next-hop node. When it receives a custody signal back for this bundle, it can use the EID to which the signal was sent to determine to which copy of the bundle the signal refers. This technique of distinguishing the multicast bundle copies from each other is conserving of both bandwidth and protocol complexity. Ideally, it would be defined as part of the multicast EID naming scheme and integrated with the routing protocols to the extent that only one registration per node would have to be propagated because only a certain portion of the EID would be used to route to each node.

D. *Forwarding Failures*

In the same way that there is a delay between when a node registers and when that registration propagates through the network to graft that node onto the distribution tree, there may also be a delay between when a node de-registers with a multicast EID and when that de-registration propagates through the network to prune that node from the distribution tree. As a result, a situation may arise in which a de-registration request has reached some nodes but not others such that a bundle could be forwarded by a custodian (which has not yet received notice of the de-registration request) and later received at some downstream node (which has received the de-registration request) that does not have any next-hop nodes to which the bundle should be forwarded (because the node that recently de-registered was the only node downstream of this node that had been in the multicast endpoint). In this situation, the bundle cannot be forwarded because there is no known route to its destination. If the node at which this situation occurs has recorded the fact that it received a de-registration notification for the multicast EID in question, it can distinguish this routing difficulty from other types of routing failures. Instead of sending a "Failed" custody signal, the node that cannot forward the bundle because a downstream node has recently de-registered from the multicast EID should send a "Succeeded" custody signal for this multicast bundle. A "Succeeded" custody signal indicates that there are no remaining copies of the bundle downstream of this node that need to be either delivered or taken custody of.

VI. Conclusions and Future Work

We have outlined our current work-in-progress for defining Bundle Protocol extensions for supporting custodial multicast delivery within DTNs and we have discussed some of the unique issues that arise in this area of work. A major building block in providing custodial multicast delivery support, definition of a multicast routing protocol for DTN, has yet to be accomplished, but we have enumerated several requirements that such a routing protocol must meet in order to support the custodial multicast delivery extensions that we are proposing. These extensions have yet to be implemented as a proof-of-concept, and their security aspects have not yet been addressed. Mechanisms will also need to be defined to meet the third custodial multicast objective listed, which is that of enabling delivery of bundles to late-registering nodes. The mechanics of how registration requests propagate to all appropriate custodians, possibly to include a method for indicating which bundles have and which have not been forwarded to the new node, have yet to be defined. In these and other aspects, defining support for custodial multicast delivery in DTNs remains a challenging, yet potentially fruitful, area of research.

References

[1] Cerf, V., et al, "Delay-Tolerant Network Architecture", draft-irtf-dtnrg-arch-05.txt , October 2005, <draft-irtf-dtnrg-arch-05.txt>.

[2] Scott, K. and S. Burleigh, "Bundle Protocol Specification", draft-irtf-dtnrg-bundle-spec-05.txt , July 2005.

[3] Symington, S., Durst, R., and Scott, K., "Delay-Tolerant Networking Custodial Multicast Extensions", unpublished.

[4] Zhao, W., Ammar, M., and E. Zegura, "Multicasting in Delay Tolerant Networks: Semantic Models and Routing Algorithms", August 2005.

[5] Symington, S., Durst, R., and K. Scott, "Delay-Tolerant Networking Bundle-in-Bundle Encapsulation", draft-irtf-dtnrg-bundle-encapsulation-00.txt , April 2006.

[6] Berners-Lee, T., R. Fielding, and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.