Networking for Communications Challenged Communities:
Architecture, Test Beds and Innovative Alliances
Contract no: 223994

D2.1

N4C System Architecture

Version 1.0

N4c-wp2-004-sys-arch.doc

Folly Consulting/Trinity College Dublin
Elwyn Davies, Stephen Farrell
www.folly.org.uk http://www.cs.tcd.ie

ABSTRACT

Starting in May 2008, N4C is a 36 month research project in the Seventh Framework Programme (www.cordis.lu/fp7). In cooperation between users in Swedish Lapland and Kočevje region in the Slovenian mountain and partners, the project will design and experiment with an architecture, infrastructure and applications in field trials and build two test beds.

This document describes the architecture for the system developed during the N4C project. The system will provide for the extension of the existing Internet network into regions where there is little or no permanent communications infrastructure and communications has to rely predominantly on opportunistic mechanisms to transfer messages during encounters between nodes carried by any available actors in these regions and be able to cope with intermittent connectivity. The aim is to provide as seamless an extension as possible, allowing users to benefit from an experience that is as close to the model available in the conventional Internet where communication is possible with delays that are commensurate with the reaction time of human beings and applications that are able to respond within the usual bounds of human expectation and tolerance, as opposed to the delays that are likely to be seen in the N4C examples which may be in the order of hours or days.

To support these requirements, the communications in the extension regions will be based on the Delay- and Disruption-Tolerant Networking (DTN) paradigm. The project will build on work done under the auspices of the DTN Research Group (DTNRG) in the Internet Research Task Force (IRTF) – primarily the DTN2 reference implementation - and during the predecessor project Sámi Network Communications (SNC) – primarily the Prophet infrastructure.

An initial version of this document was produced at a very early stage in the project. This updated version reflects the evolution of the architecture during the project and describes the progress that has been made in implementing this architecture during the project.

Due date of deliverable: 30 April 2009 Actual submission date: 10 May 2011 (Updated)

Document history			
Version	**Status**	**Date**	**Author**
1.0	Editorial updates. Delivered to EC.	10/05/2011	Elwyn Davies
0.6	Completed references and added pointers to Annexes.	09/05/2011	Elwyn Davies
0.5	Updated to reflect architecture as evolved during the project and implementation during project.	26/04/2011	Elwyn Davies
0.4	Refine node sub-system descriptions. Add inter-gateway protocol and architecture beyond the node.	04/09/2008	Elwyn Davies
0.3	Add sub-system descriptions. Also discussion of architectural issues	03/09/2008	Elwyn Davies
0.2	Implemented changes after discussions with Avri Doria. Further creation of sub-system designs.	29/08/2008	Elwyn Davies
0.1	Updated after initial comments from Avri Doria. More creation	21/08/2008	Elwyn Davies
0.0	Created	21/07/2008	Elwyn Davies

Dissemination level	
	Level
PU = Public	x
PP = Restricted to other programme participants (including the Commission Services).	
RE = Restricted to a group specified by the consortium (including the Commission Services).	
CO = Confidential, only for members of the consortium (including the Commission Services).	

CONTENT

1. INTRODUCTION

The aim of the project is to research and develop software based on the Delay- and Disruption-Tolerant Networking (DTN) paradigm that will allow extension of the existing 'Connected Internet' (CI) into regions where there is little or no existing infrastructure, and it is economically or practically unfeasible to extend the existing wired or wireless infrastructure into these areas. The software architecture has been designed around a number of flexible interfaces between a set of modules or sub-systems allowing modules to be developed independently. It is not known at the outset either what the most appropriate forms of the interfaces or what algorithms will prove most effective on the various modules: N4C aims to research and provide extensive trials of various alternatives to identify the best available methods. This architecture is intended to facilitate experimentation whereby alternative versions of modules can be developed and plugged into the architectural framework in such a way that the field trials of the software can examine the usability and performance of multiple combinations of components.

This document is divided into three major sections:

Part 1 (Sections 2 to 7)
: describes the software architecture of the DTN infrastructure as used in N4C;

Part 2 (sections 8 to 17)
: describes the major achievements of the work done in N4C relating to the architecture, provides an evaluation of this work and recommends some areas of future research and development that have been identified by the work; and

Part 3 (Sections 18 and 19)
: provides a Glossary and References for this document.

The software architecture as described in Part 1 covers the architecture of the infrastructure components that support the DTN-aware applications that N4C is developing, together with the interfaces presented to these applications. The internal architecture of these applications is outside the scope of this document, but will be described in documents applicable to the individual applications.

The node architecture builds on the architecture of the DTN2 reference implementation [DTN2] of the Bundle Protocol [RFC5050] that is coordinated by the Internet Research Task Force (IRTF) DTN research group [DTNRG]and also the Prophet DTN implementation [ProphetDTN] originally developed for the predecessor SNC project. Both of these pieces of software are being improved and added to during N4C. This work is documented in Part 2.

PART 1:
DTN INFRASTRUCTURE SOFTWARE ARCHITECTURE

2. SCOPE OF THE PROJECT

The N4C architecture is intended to allow the extension of the 'Connected Internet' (CI) into regions with little or no existing infrastructure generically known as (communications) challenged regions (CCR). Each CCR interlinks with the CI through one or more gateways that have connectivity into the CI on one side and are able to send and receive messages in a single region of challenged connectivity on the other side. It is not currently envisaged that challenged regions will communicate directly; communications between challenged regions will be expected to pass through the CI. Consideration may be given to whether it is possible to have gateways directly linking two CCRs at a later date.

The N4C project is primarily intended to research and demonstrate the applicability of the DTN paradigm as the basic infrastructure for CCRs. Accordingly CCRs will normally be known as DTN regions in this architecture. This does not preclude the use of technology that allows communications that provide links with delays similar to those seen in the CI, where communications delays are dominated by speed of light propagation delays and processing delays in a small number of intermediate routers. However communications in the DTN Regions for which N4C is targeted are expected to encounter delays ranging from a few minutes to hours or days. Furthermore connections will be intermittent.

Nodes that can operate in a CCR region using DTN need to expect to be isolated and disconnected from all other nodes for a large fraction of the time they spend within the CCR. To that extent they are more 'self-contained' than nodes that expect to have continuous connectivity to the CI. For applications to be usable when the node is in a CCR they need to be able to continue to operate without network connectivity and during (re-)connection and disconnection both from CI areas and during ad hoc encounters with other similarly CCR-capable nodes.

Message forwarding on DTN Regions will rely on opportunistic mechanisms; nodes that are intended to operate in DTN regions are generally expected to support the DTN extended store, carry and forward paradigm for 'bundles' as described in the DTN Architecture [RFC4838]. Many of these nodes will be mobile, carried for example by humans, animals and helicopters, and equipped with short range wireless connectivity such as Wi-Fi (802.11). When two of these nodes come within the range of their wireless connectivity they will be expected to detect this event and set up an exchange of bundles controlled by the routing protocol in use in the DTN Region. Because the nodes are not in continuous contact and the duration of any one contact is likely to be relatively short and of unpredictable length, end-to-end communication is subject to arbitrary delays and may be effectively disrupted if a pair of nodes are not able to exchange all the bundles that the routing protocol indicated should have been exchanged.

In addition to the CCRs linking with the main body of the fully connected CI, it is envisioned that a CCR may contain one or more isolated 'enclaves' where CI technology can be used for communication by nodes that are permanently or temporarily within the enclave. Some nodes are expected to be capable both of connecting to the CI infrastructure and making use of the CCR capabilities. Many such nodes will be mobile and a major target of the N4C project is to make the

transitions between episodes of CI connectivity, whether in an enclave or in the main body of the Internet, and CCR connectivity as seamless and transparent to the user as possible.

The overall scenario which the system architecture is intended to address is outlined in Figure 1.

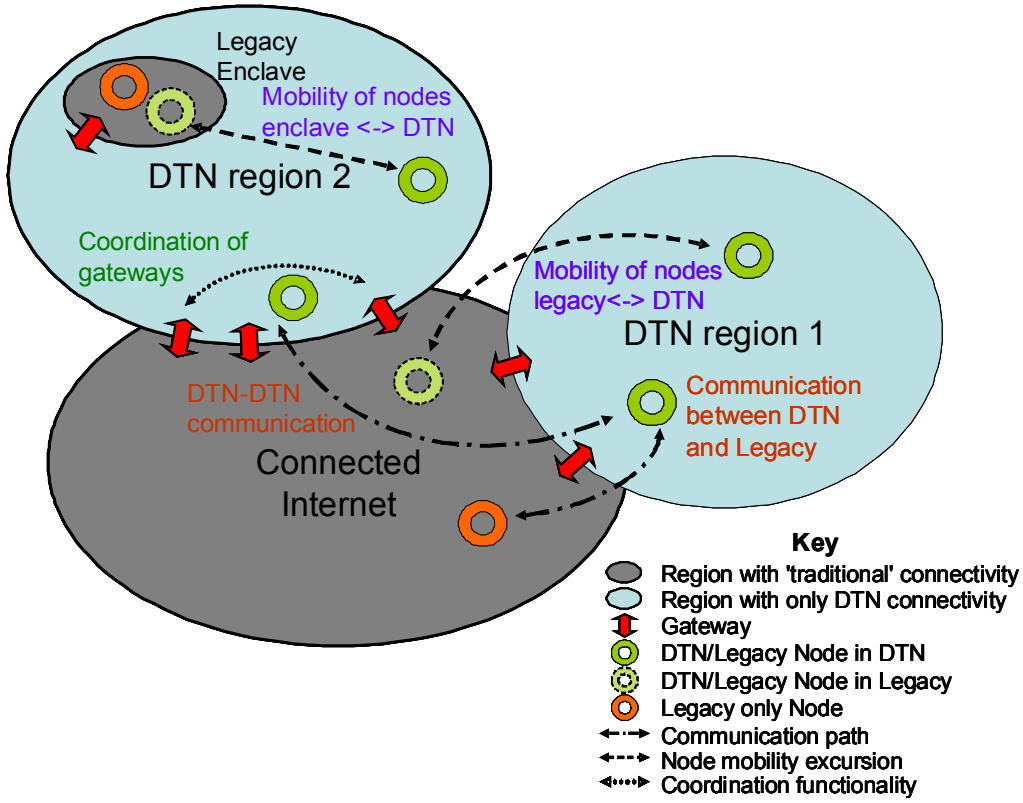

FIGURE 1 SYSTEM SCENARIO

The three types of region addressed by the N4C architecture will be referred to as **realms**.

The design of the architecture supporting the scenario illustrated in Figure 1 has to consider

- provision of connectivity for nodes in
  - the 'traditional' or Connected Internet (CI) realm,
  - realms in CCRs that can only provide DTN connectivity for most of their area but partially overlap with one or more gateways interlinking the realms, and
  - realms that are enclaves within a CCR where there is localized capability for communication using the usual CI protocols at the IP (Internet Protocol), transport and lower layers but no connectivity to the core of the CI using these protocols;
- enabling and routing communications between pairs (and possibly groups) of nodes where the communicating nodes are situated in any of the realms, whether they are all in the same realm or distributed in different realms, including the case where the nodes are situated in disjoint CCR realms and the communication has to transit the CI;
- providing support for nodes that are mobile and may be required to function in any one of the realms depending on their location; and
- coordinating the interlinking gateways to ensure that messages are routed between realms in the most efficient and appropriate way.

A major difference between the architecture of the CI and a DTN is that the DTN operates on a 'store, carry and forward' paradigm, where intermediate nodes operate in some ways like routers in the CI, serving to router and forward messages, but unlike the CI all nodes in a DTN must be capable of storing and (if mobile) carrying messages for an extended period of time. This may include taking 'custody' of the message, which means accepting responsibility for delivering the message from the previous custodian. In the CI routers will only queue messages for a short period and custody of messages always remains with the original sender until the message is delivered.

The N4C project is investigating both the infrastructure needed to support as seamless an integration of the CI and the CCRs and a number of applications that will be used to demonstrate both the technical feasibility of the DTN paradigm and the economic viability of the proposal. The following sections outline both the general nature of the applications that will be implemented and the specific applications envisaged by N4C together with the generic functionality needed from the infrastructure to support these applications. Before looking at the functionality of the applications and the infrastructure, we present the expected use cases for node behaviour that will be supported. This will have an impact on both the necessary functionality and the requirements placed on node hardware.

2.1 ELASTICITY IN THE ROUND TRIP BOUND

In the CI, applications generally have an expectation of the length of time that it will take for a remote node to respond to application messages that are sent. If the network is congested or the remote node is not responding for some reason, then after a while the application should determine that a response is not forthcoming and take appropriate action. This length of time is the Round Trip Bound (RTB). Both the usual CI applications and some CI transport protocols (e.g., TCP) expect and tolerate a certain amount of variation in the length of the round trip, but neither class can handle networks that have very large and very variable RTBs. Depending on the infrastructure involved, there will be a limited amount of variability in the expected RTB needed for working in particular circumstances on the CI. Applications that can handle this situation will be described as coping with low elasticity in the RTB (typically from fractions of a millisecond to a few tens of seconds). On the other hand, applications and transport infrastructure that expect to operate in a DTN environment need to cope with a much higher elasticity in the RTB (up to several days). The capabilities will be referenced as LERTB and HERTB. This matter is discussed further in Section 2.4.

2.2 THE NATURE OF ADDRESSING REALMS

We observe that the CI and CCRs have a fundamental conceptual difference: the core CI is a *topologically defined* region whereas CCRs that will be addressed by the N4C project are predominantly *physically or geographically defined* regions. Participation in communications within these realms is mediated in each case by a form of membership. Membership in each case is defined by having an appropriate token: an IP address for the CI and (probably) a DTN URI of a specific form for a CCR. In each case some level of authorization and authentication will be needed in an operational environment to assure communication partners that the membership is valid.

On the other hand, 'attachment' to or presence in a realm is determined by very different criteria for the two cases. A node is (usefully) present in the CI if it has low latency connectivity to another node in the CI (and hence transitively to all other nodes currently attached to the CI that are willing to communicate), whereas it is (usefully) present in a CCR by virtue of its location provided that it is aware of the existence of the CCR and has been given authorisation to participate in the operation of

the CCR concerned giving it authority to have an address of the right form[1]; it is expected that a node in a CCR will have no connectivity with any other node for a large fraction of the time that it is in the CCR.

A node is therefore usefully present in the CI if it has usable routes to other nodes in the CI, allowing packets to be despatched out of the node essentially immediately after generation (subject to short term congestion constraints). Reachability of other nodes in the CI is determined experimentally but nodes can ascertain if a potential communication partner can be reached within a short period. Communication will only be embarked on if the partner proves to be reachable within this timeframe, but the node can rely on the resources of the accessible CI to assist in determining where the partner is and how to reach it. An application running in this environment can pretty much deliver 'instant gratification' or report that communication is not possible.

By contrast a node in a CCR is very much on its own. If it wishes to communicate with a partner, it needs to have cached sufficient information to be able to address that partner without recourse to external lookup mechanisms such as the DNS infrastructure. Applications running in this environment cannot expect to deliver 'instant gratification'. They must expect to despatch units of data into the 'network' with no fixed expectation of when or if a response will be received.

CCRs are not necessarily geographically or physically bounded (one might ask if the Interplanetary Internet is physically bounded?) but physical location is key to the operation of the sort of network envisaged by N4C for CCRs. N4C is addressing CCRs that are characterised by communication through opportunistic encounters. An opportunity arises when two nodes come into communication range: here the location aspect is relative - the absolute location of the encounter is immaterial. The other constraints on the capabilities of nodes, such as power, portability and lack of communications infrastructure, conspire to limit the distance between nodes during an encounter in most cases[2]. There is no point in a node being a member of a CCR if it will not have fairly frequent encounters with other members. Furthermore there are scalability concerns if the membership grows too large as member nodes might be faced with carrying large amounts of data with relatively low probabilities of being able to deliver it together if there is a large community of possible communication partners. When combined with the expectation that a communication within a CCR will address the needs of a particular human social grouping, these constraints lead to the expectation that a CCR as envisaged by N4C will correspond to an absolute geographical or physical region.

The expectation of physical bounds on a CCR feeds into the scalability and efficiency of a DTN-based communication solution for nodes in the CCR. It is also important (although not necessarily critical) when a node outside the CCR wishes to originate communications with a mobile node that might currently be either in the CI or in one of several different CCRs. If the node is not currently present on the CI, the messages need to be directed to the CCR where the node might be able to take delivery of the message. Inevitably, if the destination node is a Fully Mobile Node (FMN) with membership in several CCRs and the ability to operate in the CI, this decision is a probabilistic rather than an absolute one, and there is a chance that the node will have moved to a different CCR by the time the message has been forwarded into the original CCR because of the extended delays that may be encountered. However, it appears that it is useful for an FMN to be able to keep track of which

[1] For example URI's with an embedded Cryptographically Generated Address (CGA) could be used to define membership. See the Secure Neighbor Discovery mechanism standardized for IPv6 [RFC3971] and [RFC3972].
[2] However it is also possible for two nodes that are members of a CCR to connect as if they were in the CCR but using the CI for communication if they are both attached to the CI at the time. This is a situation that may arise quite frequently for mobile nodes.

CCR it is currently 'in', both to allow it to inform CCR gateways of its most probable location and to allow it to direct messages to other FMNs most appropriately.

An FMN would be able to gain some insight into its location either through encounters with static nodes that are bound to a particular CCR or by using external means such as GPS, given that it is aware of the bounds of the CCR. Gateways could be made aware of the knowledge either dynamically or by administrative management.

2.3 NODE USE CASES

Nodes that can exploit some or all of the functionality of the combined CI and DTN Regions fall into a number of classes. The taxonomy of these classes is dependent on the expected range and degree of mobility of the node combined with the roles which the node can play in forwarding messages within and between CI and DTN Regions.

2.3.1 Pure Legacy Nodes

Nodes that are not capable of using the DTN functionality directly and cannot fully support adapted applications because they do not have the requisite application programming interfaces (APIs). In essence these would be 'standard' nodes normally used on the CI. As well as operating in the CI, these nodes could be used in enclaves in CCRs but would be restricted to existing applications (such as email) that can operate unchanged in a CCR enclave or using only resources within the enclave together with any adapted applications that offer a 'backwards compatibility' mode whereby they can provide all or part of their functionality when provided with a LERTB environment. It is important that Pure Legacy nodes do not require any alterations to their infrastructure (operating system and protocols) to allow them to operate in the enclaves of CCRs. However it is expected that new applications may be installed on such nodes to allow them to make best use of the DTN connectivity between the enclave and the core CI[3].

2.3.2 DTN Only Nodes

Nodes that are expected to remain purely within a DTN region outside of an enclave and do not need CI connectivity. All connectivity would be with other DTN capable nodes during opportunistic encounters using the DTN infrastructure software. This sort of node might be deployed as a 'router' node in the DTN providing store and forward capabilities at a strategic point (such as a major travel route intersection) where numbers of travellers with mobile nodes might be expected to pass by but not necessarily encounter other travellers. It might also be appropriate for use with an isolated sensor station (such as a meteorological station). Such nodes might or might not be loaded with DTN capable applications depending on their role. Some might be mobile within the CCR whereas others would be static or 'moveable'. Power supply considerations would often be very significant for such nodes as they might have to operate unattended in adverse climatic conditions for considerable periods of time.

2.3.2.1 DTN Sensor Group Leaders

In some sensor-based applications (such as herd animal tracking), it is envisaged that only a subset of the nodes deployed (such as those on the 'herd leader' animals) would have DTN capability (on

[3] An interesting question is whether we should attempt to provide a 'translucent' or cached filing system that is automatically mirrored across the DTN. Such things already exist and might be adapted for the N4C scenario. [This was not done in N4C but the question has been taken up in a slightly different form by the SAIL project – see [SAIL] and Section 10.7].

account of power and cost constraints). Such nodes would have DTN capability in order to forward the data gathered from the sensor group plus some alternative capability for communicating with other Sensor Group Members. It is possible that this communication might use DTN technology but it is more likely to be an alternative wireless technology with lower complexity and/or power economy than that needed for DTN-based communication with adjacent DTN nodes.

2.3.2.2 Sensor Group Members

Nodes associated with a 'slave' sensor that can only communicate outside the sensor group through a local Sensor Group Leader. It will generally not use DTN technology, but may use some other form of *ad hoc* networking technology, such as MANET (Mobile Ad-Hoc Network) or Zigbee (IEEE 802.15.4).

2.3.3 Fully Mobile Nodes

Fully Mobile Nodes (FMNs) are nodes that are capable of operating anywhere in the network area covered by the main body of the CI or a DTN region including in the enclaves. These nodes have to behave differently depending on the region where they are operating, but should do so in a manner that is as transparent to the user as possible. They also need to be able to sense the environment in which they are operating: because of this ability to sense and function in multiple environments, FMNs have a chameleon-like ability to adapt their behaviour to their network environment. In the CI or in CCR enclaves FMNs would be using standard Internet protocols to communicate with Pure Legacy nodes or other FMNs that have recognized that they are in an area where CI communication is possible, whereas they would use DTN mechanisms to communicate with FMNs or DTN only nodes encountered opportunistically in the CCR outside an enclave. A key area of research is how to make the transition between these modes transparent to users and to applications insofar as they are not concerned (some applications may wish to explicitly alter their internal behaviour when connected through a CCR rather than the CI, and some intermediate behaviour may be appropriate when in an enclave). To this end it may be useful to provide a common API that will allow an application to work over whichever transport infrastructure is appropriate to the realm in which it is currently working. This requires investigation and research because of the different semantics of DTN and CI transports; this is an item for future study.

2.3.4 Gateway Nodes

Because of the different characteristics of communication in the CI, CCR enclaves and CCR regions outside the enclaves, it will be necessary to provide gateways

- on the boundaries between the CI and the CCR regions where DTN protocols are used, and
- on the boundaries between the CCR enclaves and the rest of the CCR regions.

Each gateway will have one or more logical pairs of interfaces. One half of each pair will interface using CI protocols and the other half will interface using DTN protocols. The gateway will provide the mechanisms needed to mediate communications linking nodes in disparate regions and passing between these paired interfaces. Depending on the transport mechanisms employed in the DTN region, gateways may need to provide a range of services, including:

- translation between packets and bundles in both directions,
- providing custody services for information being injected into DTN regions,

- supporting FMNs moving between the CI and CCR regions,
- managing address translation,
- providing routing services for messages transiting the gateway, and
- coordinating services for a pair of regions that have multiple gateways.

It is anticipated that some gateways will have a static, dedicated role especially at the boundary between the CI and the DTN parts of CCR regions. Such nodes will have a very clear role as gateways. Gateways will also be required at the boundaries between the DTN part and the conventional enclaves of CCR regions when pure legacy nodes have to be supported in the enclave. Whilst these gateways could also be dedicated nodes, it might also be possible that FMNs could also function as gateways, since their ability to function in both environments implies a type of gateway functionality.

This architecture will not directly support gateways linking different CCR region; instead communications between CCR regions will be expected to transit the CI. However, such a gateway could be synthesised using back-to-back gateways CCR-CI gateways.

A major concern for gateways will be managing the reachability of nodes. This is especially true of FMNs, where gateways need to be aware of inter-region moves (CI⇒DTN, DTN⇒CI) made by nodes, but gateways also need to be aware of what nodes of all kinds might be reachable within a DTN region and any changes in this set due to operational events.

For the purposes of the experiments to be performed during N4C, this gateway functionality will be performed by application layer proxies that manage the transformation between packets and bundles and handle the necessary address mapping. Experience gained with proxies and further research into the aspects of gateway functionality may allow a more generic gateway functionality to be integrated into the DTN infrastructure in future.

2.4 APPLICATION FUNCTIONALITY

The intention of the N4C project is to create and demonstrate a number of useful applications during six sessions of field trials in two test bed areas. These applications will be running on top of DTN infrastructure that will be developed, enhanced and deployed during the project, and in some cases may be designed to function in both CI and CCR regions. The applications will be designed both to exercise the capabilities of the infrastructure and to provide information that will help in designing the business models that are also deliverables of the project.

The applications developed by N4C are intended to provide functionality that provides a user experience that is as similar as possible to comparable applications that are normally used on the CI. Many applications in use on the CI rely on communications round trip times between the end points being significantly lower than human reaction times, or at least less than human expectation and tolerance thresholds. Such applications rely on the CI having a relatively small (upper) Round Trip Bound (RTB). The CI with its existing applications is mostly intolerant of sub-networks that have a highly 'elastic' (variable or poorly defined) RTB; it could be said to be an environment in there was low elasticity of the RTB (LERTB). Applications running in CCRs cannot necessarily expect to experience round trip times that meet the criteria for a low elasticity RTB network. DTN infrastructures are specifically expected to potentially have much longer round trip and end-to-end

delays as well as potential disruption leading to seriously delayed or non-delivery, and can be described as high elasticity RTB (HERTB) environments.

Some applications inherently expect that there will be delays in transmission (for example, email) and need little or no adaptation of the user experience, but others rely to a greater or lesser extent on low delay real time communication between end points. For these applications, it will be necessary at least to adapt the user interface and the interaction with the remote end points to provide a usable user experience. In some cases (for example real time telephony) this may not be possible; some applications cannot be adapted to be delay tolerant and still meet reasonable user expectations. Where an application, whether adapted or not, is used in an environment where delay or disruption is possible, some management of user expectations will be necessary to ensure that users understand the different environment in which applications are trying to work.

In essence, N4C will be seeking to provide effective applications and infrastructure for a HERTB environment both through the provision of appropriate infrastructure support and adaptation of user interfaces to reflect this environment whilst managing user expectations in a way that maintains confidence in the application.

2.4.1 Classes of Application

2.4.1.1 Store and Forward Paradigm

Applications that use a store and forward paradigm (the archetype is RFC 5322-based email [RFC5322]) will need little adaptation to work in a DTN environment. Provided user expectations of delays are managed appropriately, a DTN network can provide the necessary forwarding for messages with no need to modify operation of the user interface. If the DTN network offers secure custody transfer, applications should consider whether a chunk of data that is transferred to the custody of the DTN also needs to be stored pending receipt of end-to-end delivery confirmation.

Some peer-to-peer applications nominally fall into this class where they download chunks of data from multiple sources, but the dynamic nature of the connections between peers and the adaptation of loads on sources in real time is likely to be challenging in a CCR. Also a node in a CCR will not be a very responsive source peer for other users. In practice such peer-to-peer applications are closer to the client-server paradigm where requests are made to multiple servers.

2.4.1.2 Client-Server Paradigm

Applications that use a client-server or request-response paradigm will generally require significant adaptation and potentially support by proxies to provide an adequate user experience where the client is in the CCR and the server is in the CI. In very simple cases the user may just experience a long delay between sending the request and the response being displayed, but there are very few realistic cases where this applies. Applications, such as those that use the World Wide Web through the HTTP protocol, would encounter a number of challenges:

- Communication timeouts due to the expectation of standard transport protocols TCP and UDP,
- Use of recursive requests, whereby the initial response triggers one or more additional requests driven by data embedded in the initial and, possibly, subsequent responses, requiring multiple round trips between client and one or more servers,

- Dynamic construction of responses and use of cookies to construct sessions between client and server, leading to timeouts if long delays are encountered,
- Responses containing active content such as Javascript or C# that modifies the requests sent to the server depending on user input or browser configuration, and
- Interactive and especially secure interactive, applications where multiple request-response cycles are driven by user input.

Some these challenges can be addressed by using proxies. For example a proxy located in the CI could potentially manage recursive requests by triggering the recursive requests on behalf of the client in the CCR region, parcelling up the complete set of responses and passing them back to the client (or a second local proxy) that could then emulate the actual server responses when the parcel was received across the DTN region.

However applications will generally need to be modified to be aware that they are running in a DTN environment, and the infrastructure will need to be able to provide indications about the type of environment (synchronous or DTN) in which the node is currently located, and notifications if the node moves to a different environment.

For some applications, especially secure interactive applications, it may prove necessary for the server to provide alternative responses if it is made aware that the client is located in a CCR. For example the server could provide a more autonomous program in response to the initial query that gathers the necessary set of responses and parcels them up in a secure envelope with unforgeable credentials for transmission to the server in one bundle.

Locating the server of the client-server interaction in a CCR might not appear a rational move if we are considering world wide web services, but in practice, many of the applications such as animal tracking, that N4C will be considering use a form of client-server paradigm to deliver information from within the CCR.

2.4.1.3 Unidirectional Real Time Stream Paradigm

Applications that deliver a real time data stream from a source to a destination will generally not work without assistance or adaptation across a DTN connection. A proxy based solution that provides a much larger buffer may be adequate if the information in the stream is not very time sensitive (for example, a television entertainment programme) but there may be issues of size and digital rights management if the proxy intends to capture the whole stream and pass it back to the client as a single bundle[4]. Consideration needs to be given to the case where a node receiving such a stream moves from the CI to the CCR.

2.4.1.4 Interactive Real Time Stream Paradigm

This sort of application will not be usable across DTN regions. However, it would be useful to consider providing notification to users if a node moves into the DTN during an interaction. Making the application aware of the possibility that the streams would be interrupted would improve the user experience.

[4] The implications for Internet governance and the interactions with copyright law need to be taken into consideration. This aspect should be considered as part of the business modelling and societal impact portions of the N4C project.

2.4.2 System and Management Applications

In addition to user applications a number of management applications will be developed to aid configuration, monitoring, and analysis of the DTN applications and infrastructure.

2.4.3 Interface with Infrastructure

A number of Application Programming Interfaces (APIs) are provided to allow applications to use, control and monitor the DTN infrastructure.

Interfaces will allow appropriately authorized applications to

- Determine when DTN capability is available in a node
- Determine the modes of communication currently available (DTN bundle protocol, direct connection to CI, etc)
- Open, manage and close communication channels using DTN protocols providing control appropriate for both end user applications and proxy servers
- Send and receive messages over the DTN infrastructure
- Receive notifications when the modes of communication alter due to movement or reconfiguration of the node
- Control and use security mechanisms to allow authentication and encryption for messages
- Control and monitor the addressing of the node
- Control and monitor DTN routing mechanisms
- Control and monitor message (packet, bundle, etc) storage and forwarding

2.5 INFRASTRUCTURE FUNCTIONALITY

Since N4C is intended to be an experimental project, it is envisaged that several different implementations of modules of the infrastructure functionality will be developed, deployed and tested during the project. It is therefore essential that the infrastructure is highly modular with well defined interfaces and mechanisms that allow modules to be replaced easily, preferably under remote control, and without the need to return the node to a facility connected to the CI.

In order to support the application functionality described in Section 2.4, the DTN infrastructure needs to provide a number of pieces of functionality: For the purposes of N4C the following functionality will be needed:

- Forwarding of DTN bundles as specified in the Bundle Protocol [RFC5050]
- Support for DTN convergence layers, including at least TCP, between adjacent nodes
- Routing mechanisms suitable for directing DTN bundles through the CCR
- Support for authentication of nodes and messages in the DTN environment

- Support for encryption and integrity checking of messages passing through the DTN environment
- Manage configuration and addressing of nodes providing DTN functionality
- Manage the power consumption and availability of node functionality to match the available power in the expected deployments of DTN nodes where energy input is strictly limited.
- Logging and monitoring of the DTN infrastructure. For this purpose as well as the correct operation of the Bundle Protocol and Power Management requiring a diurnal cycle a reasonably accurate real time clock must be available providing wall clock time including after any shutdowns of the main node systems as may occur if there is a power shortage.

It is anticipated that in future further integration of the CI and DTN realms will require additional functionality which will not be implemented in the N4C infrastructure, but will be taken into consideration during the design of systems. Some architectural research relating to these areas will also continue and will be reported both in N4C and in the DTN research group. This functionality includes:

- Handle mapping of addresses between the CI and DTN forms where necessary
- Gateway mechanisms (in place of application level handling by proxies)
    - to mediate inter-realm communications between the core CI and CCR DTN regions
    - to mediate inter-realm communications between CCR DTN regions and CCR enclaves using CI protocols
    - manage linkages between realms where multiple gateways are provisioned.
- Manage mobility of FMNs moving from CI realms to DTN realms and vice versa
- Mechanisms for secure distribution and installation of new components over the DTN infrastructure, and for full remote configuration and management of nodes through the DTN infrastructure.

2.6 PHYSICAL NETWORKS

N4C will make extensive use of wireless technology in the CCR, with possibly limited use of wired technology (probably Ethernet) in CCR enclaves. In line with the SNC predecessor project, N4C expects to use Wi-Fi (802.11) technology as the main means for exchange of data between nodes during opportunistic contacts during initial experiments and in CCR enclaves.

N4C will also experiment with other radio technologies such as WiMAX (IEEE 802.16), Nordic Mobile Telephone 450 MHz (NMT 450) and possibly ZigBee (IEEE 802.15.4) both as a means of extending the reach of N4C technology and to deal with the challenges stemming from the need to minimize power consumption on mobile nodes in regions where there is no mains power supply and renewable sources of energy are less productive (for example, in the Sámi arctic field trial area solar energy is not an option during the depths of the winter and even during summer, cloudy weather can seriously limit the availability of solar energy).

N4C is using opportunistic contacts between nodes in moving vehicles, especially helicopters, as a means of transferring data. Extending the range of contacts might allow for contacts to be effective during fly- or drive-bys as opposed to just during landings/stops, thereby improving the utility of the vehicle mobile nodes.

The project will also investigate ways of extending the range and types of terrain in which technologies such as WiMAX could be used. The topography of many CCRs is such that line-of-sight communications are impossible without the installation of expensive and intrusive antennas on high points. Lowering the frequency might allow communications even across intervening landscape features: DTN technology would still be desirable because communication may be disruption prone and nodes in the CCR would be power limited so that continuous communication would not be feasible[5].

[5] In practice lowering the frequency is not a realistic option because the size of antenna needed at the lower frequencies makes deployment difficult, as demonstrated by the results from Work Package 6 in N4C.

3. SUB-SYSTEMS

The N4C architecture is intended to be an enhancement of the conventional architecture used to support applications that work in distributed nodes linked by an Internet communications infrastructure and use the communications infrastructure as an integral part of their functionality. The major parts of the system that will be enhanced or introduced as part of the N4C project are:

- Node infrastructure and applications programming interface enhancements to support DTN-based communications (see Sections 3.2.1);
- Routing mechanisms and protocol infrastructure to support routing within the HERTB (DTN) realms (see Section 3.2.2);
- Interface between the BPA and the platform network interfaces known as Convergence Layers (see Section 3.2.3),
- Persistent storage management (see Section 3.2.4),
- Power Management systems to ensure that the node can operate in a way that makes maximum use of the available power, entering and returning from low power consumption 'sleep' modes according to what is available in the hardware platform (see Section 3.2.8); and
- Management and monitoring capabilities for the enhanced system (see Section 3.2.7).

3.1 NODE INFRASTRUCTURE

All nodes that support N4C DTN functionality will use a set of common modules to deliver the required infrastructure functionality. Not all modules will necessarily be installed or activated in all nodes: the selection of active modules will be dependent on the class of node (see Section 2.3).

As noted in Sections 1 and 2.5, the infrastructure functionality needs to be highly modular. In future it is expected the infrastructure will need to provide the ability to dynamically replace and reconfigure modules, with replacement modules delivered over the network, including over the DTN network. The sub-systems needed to deliver the infrastructure functionality for N4C and their inter-relations are outlined in Figure 2 and described in more detail in Section 3.2.

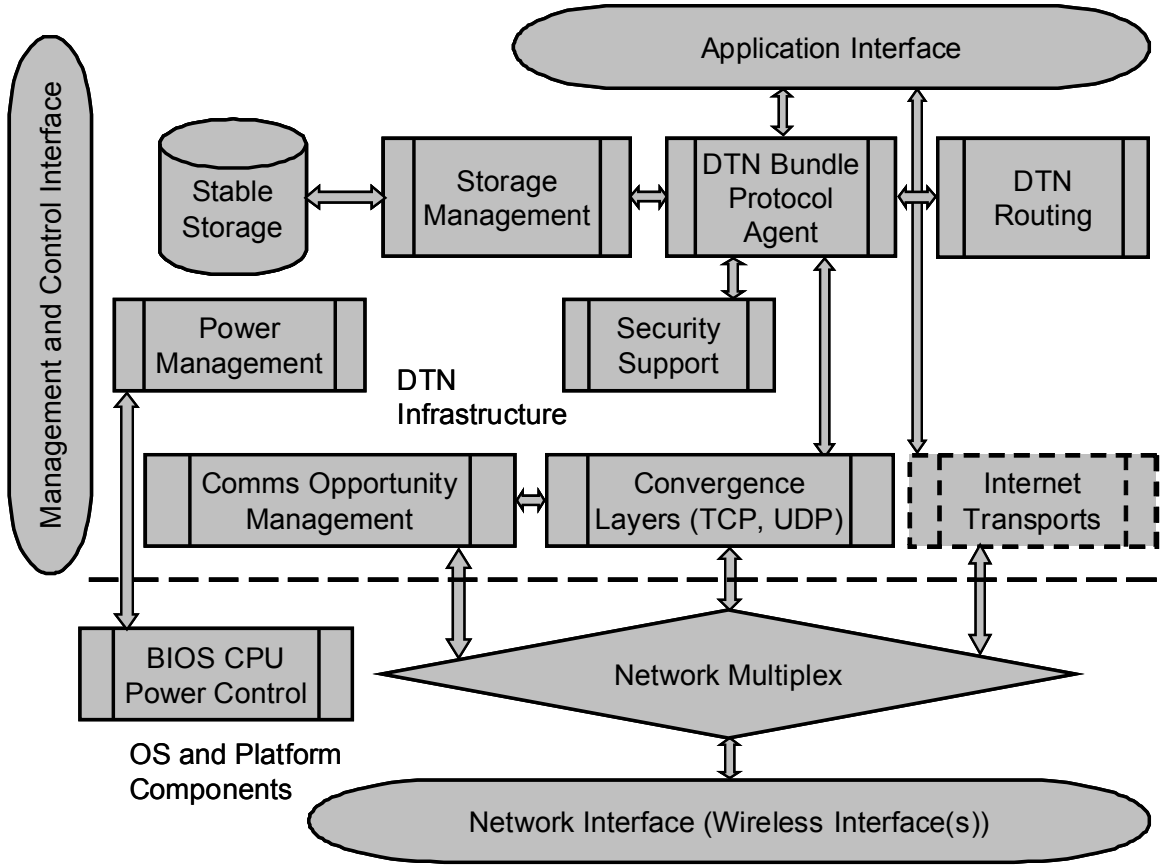

FIGURE 2 NODE SOFTWARE SUB-SYSTEM STRUCTURE

All sub-systems will have management, monitoring and logging capabilities that can be accessed through the management interface. Because DTN nodes operate independently but will be sending bundles from node to node, it is particularly critical for logging to record accurate time stamps and globally unique identifiers for bundle events and the associated bundles to allow the progress of bundles to be tracked through the network. The issue of the use of absolute time values in DTN is a matter for ongoing debate and research; indeed, time is in many ways a much more significant issue in a DTN realm than in the CI. This matter is discussed further in the reports of the evaluation of the architecture later in this document.

Section 3.2 contains descriptions of each of the sub-systems in terms of its purpose, its functionality, its interfaces with other sub-systems, and its component parts. It is anticipated that for the most part of the N4C project will be implemented in software and hardware components that are generally expected to be off-the-shelf items with already standardized interfaces and software drivers. This architecture deals almost entirely in software components. Hardware components are relevant only in so far as their characteristics (such as bandwidth, activation, knowledge of available adjacent stations) may need to be taken into account by the software components. The major influence here will be the type of radio link layer(s) that are available to a node. The kinds of physical and link layers expected to be used by nodes in N4C are listed in Section 4.2.

The components to be created or adapted by N4C will be supported by pre-existing operating systems of various kinds that will supply basic platform management, such as process and memory management plus implementations of CI protocols, file systems, display drivers, system services, etc. The software components will be designed and implemented so far as is possible in a platform and

operating system independent manner, so that components can be used on a variety of platform with little or no adaptation. That being said, the primary implementation target of the N4C components will be the Linux operating system..

So far as is possible, the N4C software components will offer or extend existing standardized interfaces to applications. Similarly, the N4C software components will make use of existing standardized interfaces to preexisting system components wherever possible.

3.2 NODE SUB-SYSTEM SPECIFICATIONS

The core DTN sub-systems will be based on the DTN2 reference implementation in the first instance. The relevant sub-systems are:

- DTN Bundle Protocol Agent (Section 3.2.1)
- DTN Routing (Section 3.2.2)
- Convergence Layers (Section 3.2.3)
- Storage Management and Stable Storage (Section 3.2.4)
- Communications Opportunity Management (Section 3.2.5)
- Security Support (Section 3.2.6)
- Management and configuration mechanisms (Section 3.2.7)
- Power management (Section 3.2.8)

Other implementations may be made either of individual sub-systems or the complete node infrastructure. Detailed specification of the functionality of this part of the node system will be given in N4C Deliverable 2.2.

3.2.1 DTN Bundle Protocol Agent

3.2.1.1 Purpose

To implement the DTN Bundle Protocol Agent (BPA) as specified in [RFC5050] together with auxiliary operations required to manage the auxiliary functions (DTN routing, bundle storage, convergence layers, communications opportunity management and security support).

3.2.1.2 Functionality

According to RFC 5050, a BPA must offer the following services to applications:

- commencing a registration (i.e., indicating that this endpoint (node) will take delivery of suitably addressed bundles);
- terminating a registration;
- switching a registration between Active and Passive states;
- transmitting a bundle to an identified bundle endpoint;
- canceling a transmission;
- polling a registration that is in the passive state;
- delivering a received bundle.

The nature of these operations is described in more detail in RFC 5050.

The BPA sub-system also handles

- configuration and management of the BPA, including remote management from other nodes and ensuring that all relevant information relating to the state of the BPA and the registrations established by applications are maintained in stable storage so that the BPA can be shutdown and restarted without loss of information,
- the encapsulation of application data for transmission on behalf of applications,
- delivery of bundles to applications in accordance with the destination node locators (known as Endpoint Identifiers or EIDs) together with any demultiplexing information registered by the application (using *service tags)*,
- management of the resources of the node that are made available for DTN operations including the temporary and permanent storage for bundles needed to implement the 'store, carry and forward' paradigm of DTN through the Storage Management sub-system (Section 3.2.4),
- events resulting from discovery of and loss of links to other DTN capable nodes providing connections over which this node is able to exchange bundles with its peers through the Communications Opportunity Management sub-system (Section 3.2.5),
- management of the routing information needed to control the forwarding of bundles on the connections established to DTN capable peers including exchanging routing information needed for dynamic routing protocols using the DTN Routing sub-system (Section 3.2.2),
- management of security functions and keys needed to provide authentication, integrity checking and encryption as requested by applications and node policy using the Security Support sub-system (Section 3.2.6), and
- control of forwarding of bundles to connected DTN capable peers in accordance with the routing information appropriate for that connection and seeking to make optimum usage of the resources of this node using the Convergence Layers (Section 3.2.3).

3.2.1.3 External Interfaces

The main external interface of the BPA sub-system is the Application Protocol Interface (API) which provides the following functions to be used by applications that which to send and/or receive bundles:

- Open a channel from application to BPA. Returns an opaque handle identifying the channel. This channel handle is used as a parameter for all other functions in the API to link the channel to the associated state in the BPA.
- Close a previously opened channel using the handle.
- Return the last error type from any of these interface functions.
- Force the error type returned by the previous function.
- Construct a DTN address (URI) from the node EID configured into the BPA and an application supplied service tag. Return an ASCII string representing the URI.
- Register a DTN address with the BPA: bundles received with a destination address matching the registered address will be handled according to parameters specified with the registration. The address must be an address in the space of URIs known to the BPA (i.e., constructed from the node EID and a service tag). Registrations may be created either in an active or passive state. In the active state registrations are bound to a currently open application channel. Passive

registrations may or may not be bound to an application channel. A lifetime is specified for the registration when it is created; this may be much longer than the lifetime of the creating application. The BPA maintains the registration state until the lifetime expires or the registration is explicitly deleted. Received bundles matching the registration are either queued for delivery through the bound channel if the registration is active when they are received or may be treated in various ways specified when the registration is created if the registration is passive; the bundle may be specified to be immediately dropped, to be stored for deferred delivery when the registration becomes active or a specified script may be run in conjunction with deferred delivery. The script would generally be expected to start an application that would open a channel, bind to the registration and take delivery of the deferred bundle. Deferred bundles will be deleted if their lifetime expires before any application chooses to take delivery. Return an identifier that can be used to manage the registration. If the registration is created in active mode it is automatically bound to the channel through which the registration is made.

- Unregister a previously registered DTN address using the identifier returned when it was registered. Any bundles awaiting delivery as a result of deferred delivery will be immediately dropped.
- Find a registration (if one exists) for a specified DTN address. Return the registration identifier if a registration is found.
- Bind a registration specified by the identifier returned when it was registered or found to a previously opened channel. The binding is made to the channel through which the binding is requested.
- Unbind a registration from the channel on which the request is made. Any bundles matching the registration that have been received but not delivered will be treated according to the defer or drop policy of the registration defined when the registration was created.
- Create and send a bundle. The parameters that are used to create the bundle are:
  - Source and destination EIDs
  - EID to which reports about progress of the bundle should be sent
  - Priority level as specified in RFC 5050
  - Set of reports to be generated as the bundle progresses through the DTN
  - Flag indicating if custody of the bundle should be transferred if allowed
  - Flag indicating if bundle fragmentation is allowed
  - Lifetime of the bundle
  - Payload data
  - Set of Bundle Protocol extension blocks (if any) to be attached to the bundle
  - Set of Bundle Protocol metadata blocks [MetadataBlock] (if any) to be attached to the bundle
    - If the security policy of the node (see Section 3.2.6) requires integrity protection or encryption to be applied to the bundle, the BPA will add these before sending the bundle using the security policy database to supply appropriate keys and algorithms.

  On successful sending, the function will return an identifier for the bundle. The bundle will be assembled in bundle storage (see Section 3.2.4) and maintained until a successful delivery report is received or the bundle lifetime is exceeded. Depending on the current DTN routing information (see Section 3.2.2) and the communication opportunities that arise (see Section 3.2.5), the bundle

will be forwarded to any encountered nodes that are deemed to offer a reasonable expectation of delivering the bundle to its final destinations(s).

- Cancel bundle sending. The identifier returned when the bundle was originally sent is provided as a parameter. No further attempts will be made to forward or deliver the bundle and the bundle will be deleted from bundle storage.

- Receive a previously delivered bundle. The data received with the bundle will include:
  - Source EID
  - Destination EID
  - EID to which reports should be sent
  - Priority code
  - Delivery options flags
  - Expiration time
  - Creation time
  - The count of blocks in the extension block list followed by the blocks
  - The count of blocks in the metadata block list followed by the blocks

  Additionally, if the block represents a bundle status report, the contents of the status report will unpacked into a suitable data structure. Otherwise the payload data will be made available as delivered. The application will have the choice of having the payload data as a file or a memory buffer.

The other External Interface of the BPA provides for management and configuration of the functionality of the BPA from terminal or scripts. The DTNRG management sub-group is working on a Management Information Base (MIB) that describes the items that need to be managed through this interface. In the interim, the DTN2 reference implementation provides an extensive management interface using the TCL scripting language to allow control and monitoring of the BPA and associated sub-systems.

3.2.1.4 Internal Interfaces

The BPA has interfaces to:

- DTN Routing used to determine if a bundle should be forwarded when a communications opportunity arises.
- Storage management used to store and retrieve bundles received from the network or created through the API in stable storage until they can be delivered to an application or forwarded or their lifetime expires.
- Convergence Layers used to send and receive bundles to other nodes through one or other of the available (local) transport protocols when a communication opportunity arises.
- Communications Opportunity Management used to send outgoing notifications and handle incoming notifications of the presence of DTN capable nodes in the wireless range of the node. When a notification is received or an outgoing notification receives an acknowledgement the BPA is informed and an appropriate Convergence Layer is invoked to establish a link for the exchange of bundles and other information, such as routing information, with the node that has come into range offering a Communication Opportunity. When the node goes out of range the opportunity is ended and the BPA and convergence layer are informed.

- Security Support is used to maintain a Security Policy Database for the node, including appropriate security suites (defined in [BSP] to be used for specific destinations, and keys to use for security operations on incoming and outgoing bundles.

3.2.2 DTN Routing

3.2.2.1 Purpose

Provide routing support for the Bundle Protocol Agent. This sub-system provides routing for the HERTB environment in the DTN when there are not stable connections and the topology is not necessarily stable. The N4C experiments expect to use a combination of statically configured routing, flood (epidemic) routing and PRoPHET dynamic routing.

3.2.2.2 Functionality

Whenever a new bundle arrives in the BPA, either over the network or from a local application, the currently active links to communication opportunities are inspected to determine if the bundle should be forwarded on one or more of these links. When a new communication opportunity starts, all currently stored bundles with unexpired lifetimes other than those that will only be delivered to locally attached applications, are examined to determine if they should be forwarded to the new communication opportunity. In either case, the stored routing information applicable to the active links is inspected to determine if the forwarding should occur. The nature of the information and the algorithms used to decide whether to forward are dependent on the routing mechanism configured for the link.

The DTN routing sub-system manages the information needed for each sort of routing mechanism and can be configured through the management interface. Pending the completion of the DTN MIB the DTN2 configuration mechanism provides means to configure routing.

For static routing:

- Configuration builds a set of {destination, way point} pairs with associated priority values. The routing mechanism seeks to build paths to the destination of a bundle that is a candidate for forwarding out of these pairs by recursively building a sequence of pairs in which the destination in one is the way point for the next with the bundle destination as the destination in the final pair. If the way point of the first pair in the sequence is the EID of the node with a currently active communication opportunity then the bundle is forwarded on that link. If more than one sequence is possible then either the bundle can be sent on all available paths or the priority value can be used to select the 'best' path.

For flood (aka epidemic) routing:

- Each bundle extant in the node is sent to every node that starts a communication opportunity provided it hasn't been sent before and each new bundle received or created is sent on all active links. The routing sub-system has to keep track of which bundles have been sent to the nodes that offer communication opportunities to minimize duplication which wastes bandwidth on the links and can lead to 'bundle storms'.

For PRoPHET routing:

- The PRoPHET routing protocol is being tested and further developed during N4C [PRoPHET]. PRoPHET routing is a form of 'pruned' epidemic routing. Instead of all bundles being forwarded to every communication opportunity, the node maintains a set of 'delivery predictabilities' for all the bundle destinations that it is aware of evolve over time as a result of encounters between

nodes. During encounters the current sets of delivery predictabilities are exchanged before bundle forwarding starts, and the values are combined in such a way that when the values of the delivery predictability for a bundle destination in the current node and the encountered node are compared, the relative values of the delivery predictability indicate whether forwarding a bundle is likely to enhance its chances of being delivered. If not the bundle is not offered for forwarding and bandwidth is saved and storage resource in the encountered node will not be taken up by bundles that are unlikely to get delivered from the encountered node. To manage this, the routing sub-system has to maintain the set of delivery predictabilities and handle the exchange of the sets when communication opportunities are in progress. The process is fully described in the PRoPHET specification [PRoPHET].

Additionally, for nodes that are gateways at boundaries that feature two or more gateway nodes additional information may have to be maintained to ensure that bundles are forwarded from the most advantageous node into the DTN realm. This may involve passing the bundle between gateways where it can be passed across a continuously active link in the CI or the enclave network.

3.2.2.3 Interfaces

The DTN Routing sub-system interfaces with the BPA (see Section 3.2.1) receiving notification of receipt of new bundles and links starting and finishing. The DTN Routing sub-system can access information about the bundles currently stored by the BPA in order to determine bundles to mark for forwarding on currently active links. In the case of the PRoPHET protocol the DTN Routing system has to manage the exchange of delivery predictability information with encountered nodes.

3.2.3 Convergence Layers

3.2.3.1 Purpose

To create and manage point to point links between pairs of nodes using the available transport protocols such as those from the IP suite used in the CI (primarily TCP and to a lesser extent UDP), Bluetooth protocols, and the Licklider Transmission Protocol (LTP) that was originally designed for very high delay paths in space-borne operations [RFC5325]. The DTN2 reference implementation contains a number of other convergence layer implementations including one running directly over link layer protocols such as Ethernet. It is not anticipated that these will be relevant to N4C.

Additional convergence layers can be added in future.

3.2.3.2 Functionality

The Bundle Protocol [RFC5050] does not specify exactly how bundles will be transported between nodes that support Bundle Protocol Agents (i.e., Bundle Daemons). Instead it expects that many different transport protocols will be used depending on the kind of network in which a node is situated. For each transport protocol that could be used an adaptation or *convergence layer* has to be defined that will allow bundles to be sent and received across the type of network on which the transport protocol runs (e.g., the TCP and UDP convergence layers allow bundles to be sent across an IP-based network using unicast transports, the Bluetooth convergence layer allows bundles to be sent between paired Bluetooth nodes, and the Ethernet convergence layer allows bundles to be sent between nodes in 'raw' Ethernet frames).

There are separate specifications that specify the operation of the three main Convergence Layers that are expected to be used in N4C. Of these TCP is expected to be the most frequently used.

- TCP over IP connection oriented transport – [TCPclayer]
- UDP over IP datagram transport – [UDPclayer]. Note that this is a very simplistic convergence layer and the DTN2 implementation of the UDP convergence layer is restricted carrying bundles that do not exceed the Maximum Transmission Unit of the connecting network.
- LTP [LTPcl]
- Optionally, RFCOMM connection oriented transport over Bluetooth [RFCOMM]. The TCP convergence layer specification can be used to specify the format of the protocol messages as the semantics of RFCOMM are very close to those of TCP.

The Convergence Layer sub-system uses operating system functionality to manage the transport interfaces that provide multiplexing of the network connectivity (typically the socket interface to the operating system) and create/destroy point to point links to other nodes. It is informed of new communications opportunities and creates communication links in response to the nodes offering the opportunity. For some types of transport, the convergence layer may monitor the link if appropriately configured and destroy the link if communication is no longer possible. In other circumstances the link may be automatically destroyed if it remains idle for more than a configured amount of time.

3.2.3.3 Interfaces

Convergence Layers interface with the BPA (Section 3.2.1) to send and receive encoded bundles and provide notification of new and terminated links.

Convergence Layers interface with the Communications Opportunity Manager (see Section 3.2.5) in order to be informed of new communication opportunities and what kind of convergence layers are supported by the encountered node so that appropriate links can be created.

Convergence Layers interface with the operating system provided transport protocols via the transport multiplexing mechanism. Typically this uses the socket interface.

3.2.4 Storage Management

3.2.4.1 Purpose

Bundles and other state data have to be maintained in persistent, stable storage to support the BPA (see Section 3.2.1) for two interrelated reasons:

- in case the BPA has to be stopped and restarted for any reason (such as power management or in the event of a node failure) without losing the bundles previously received and the information about what has been done with the bundles in terms of forwarding and deferred delivery, and
- to ensure that bundles for which the node has assumed custody can be resent if delivery fails for any reason.

Clearly the second aim cannot be achieved if the first one is not, as custody assumes that the node can look after the bundle until its lifetime expires irrespective of the fate of the BPA during this time.

3.2.4.2 Functionality

The Bundle Storage Management sub-system provides means to store, retrieve, delete, and modify complete bundles including payloads and all meta-information in stable storage. It should also allow state information relating to the forwarding history of the bundle to be maintained. This may be achieved using a database, operating system files or otherwise provided that the system is adequately robust in the face of both controlled and unexpected shutdown of the BPA and/or the whole node. The sub-system must ensure that the BPA is able to know when the storage quota would be exceeded by storing new information of a given size so that it can prioritize use of the storage for bundles for which it has custody and handle storage resource shortage gracefully.

3.2.4.3 Interfaces

The Bundle Storage Management sub-system interfaces with the BPA to provide the information storage functionality.

It also interfaces with the operating system to maintain its allocated storage space.

3.2.5 Communications Opportunity Management

3.2.5.1 Purpose

To manage advertisement of communication opportunities to prospective partners and receive such advertisements from other nodes. When suitable advertisements are received to notify the appropriate Convergence Layer (see Section 3.2.3) to start a link with the prospective partner through which bundles and other information can be exchanged as appropriate.

3.2.5.2 Functionality

The sub-system can be configured to advertise the node's ability to communicate using one or more Convergence Layers. The mechanism used for advertisements will generally depend on the transport protocol to be used in the advertised Convergence Layers. Thus TCP and UDP convergence layers should generally be advertised using IP protocols as both nodes will need to support IP if communication is to succeed. Similarly a Bluetooth Convergence Layer will generally be advertised using the built-in Bluetooth service discovery mechanisms [Inquiry]. The sub-system should also listen for advertisements relating to Convergence Layers that are configured to be available on the node using appropriate protocols. When an advertisement is received, it will specify the Convergence Layer offered and the identity (EID) of the offering node. Accepted advertisements are notified to the appropriate Convergence Layer to start a link with the offering node.

In future the information received here could be made available to applications in conjunction with knowledge about whether the node was currently connected to the CI in order for applications to optimize their behaviour depending on the realm in which the node was operating. This will not be done for N4C experiments but will be studied further in future research.

3.2.5.3 Interfaces

The Communication Opportunity Management sub-system interfaces with the Convergence Layers to request start of new links.

The sub-system also interfaces with the platform operating system to send and receive advertisements. For IP Convergence Layers this will normally be using multicast UDP datagrams.

Note that there is currently no formal specification of the advertisement protocol; the DTN2 reference implementation provides the next best thing to a specification.

3.2.5.4 Interfaces

The sub-system provides notification of new communication opportunities to appropriate Convergence Layers (see Section 3.2.3).

The sub-system interfaces with the operating system provided transport protocols via the transport multiplexing mechanism. Typically this uses the socket interface.

3.2.6 Security Support

3.2.6.1 Purpose

The Security Support sub-system provides authentication, integrity protection and encryption specific to the DTN environment using the Bundle Protocol as defined in the Bundle Security Protocol [BSP]. The security mechanisms used in conjunction with the Bundle Protocol were still under development during the experimental period of N4C and the BSP was completed only near the end of the N4C project. The DTN2 Reference Implementation contains an implementation of an earlier version of the specification and the necessary Security Policy Database is in a rudimentary state.

3.2.6.2 Functionality

The security mechanisms to be implemented are described in [BSP]. A Security Policy Database and key management system are provided to determine what mechanisms should be used when forwarding bundles to specific destinations and the appropriate keys to use with these mechanisms. It also provides information to identify the keys to be used to authenticate, check the integrity and decrypt incoming bundles from specific sources.

3.2.6.3 Interfaces

The sub-system interfaces with the BPA (see Section 3.2.1). Bundles originated in this node may and previously received bundles where this node is the specified as the security source (see [BSP]) will be modified to add security-related information blocks and, where encryption is required, have their payloads modified before forwarding. Thus the BPA will pass such bundles to the sub-system for modification before forwarding. The BPA will also pass bundles received that contain security-related information blocks to the sub-system for verification and, where necessary, decryption before delivery or further forwarding if this node is the current security destination for the bundle.

3.2.7 Management and Configuration Mechanisms

3.2.7.1 Purpose

The various sub-systems of the DTN infrastructure require extensive configuration and management. In the future this will carried by 'in band' management protocols over the DTN infrastructure. However at the start of the N4C project the Management Information Base (MIB) and protocols were only just starting to be developed. The DTN2 reference implementation contains an extensive configuration and management command system which operates on local or IP connections.

3.2.7.2 Functionality

The sub-system will provide necessary management and configuration mechanisms for the other sub-systems. During N4C this will be carried out through node local functions. Monitoring of the node

will be carried out by an external application that uses the management interface to extract information form the BPA and package it into bundles which will be sent to a management application through the DTN. N4C members will participate in the development of the MIB and means to do management and configuration internally within the BPA during N4C.

3.2.7.3 Interfaces

The sub-system interfaces with all other sub-systems. For sub-systems except Power Management, this will be mediated through the BPA.

3.2.8 Power Management

3.2.8.1 Purpose

In the experiments that N4C will be carrying out it is recognized that many nodes will be operating in areas that do not have mains electricity infrastructure. N4C intends to carry out long term experiments (months or even years) with systems using locally scavenged power (such as photo-voltaic collectors or wind powered generators) stored in a local battery. These systems need to monitor and manage the available power, energy inputs and outputs and usage demands in order to run the node, as far as possible, without human intervention.

Some nodes will be using the most recent generations of low power processors in their hardware platforms. These processors provide a wealth of capabilities for managing the energy consumption of the processor to match the instantaneous processing demand and to allow the processor to be set into various states that reduce energy consumption, ranging from reduced clock speeds through sleep modes to complete shutdown. N4C will exploit these states through a power management sub-system that provides a basic diurnal cycle built around a system power budget anticipating that during certain parts of the day (more specifically, the night) there will be little regular demand for communication services and the node can be placed into a very low power 'sleep' state. Additionally the system will be placed into less energy demanding modes when idle or when the battery state indicates that power supplies are limited. Ultimately the system should be able to power down the main part of the system if power levels drop to levels where the battery would be damaged by further consumption and then restart it when the battery has recharged.

3.2.8.2 Functionality

The sub-system will use inputs from battery state monitors (voltage and current drain/feed), the current time of day and a manual control override to request the processor to run in an appropriate mode via the platform BIOS interface, and, if necessary, to execute sets of commands to startup and shutdown the whole node.

3.2.8.3 Interfaces

The sub-system interfaces with the platform BIOS, real time clock and external battery monitor. It also executes control scripts to startup and shutdown the whole node. It also takes input from a manual control input that can provide override of the programmed state if an operator needs a burst of activity to handle a particular situation.

3.3 PLATFORM AND OPERATING SYSTEM FUNCTIONALITY

3.3.1 Internet Transports

Applications may also use the 'standard' IP suite transport protocols directly when in an enclave or in the CI. The application may select an appropriate transport mechanism if it is aware of the environment in which it is operating. Alternatively it may opt to use DTN transports or CI protocols for all communications. In this case, depending on the location of the node the communication may or may not succeed, or may involve considerable delay.

3.3.2 Network Multiplexing

This sub-system is typically built into the platform operating system. In current operating systems, IP protocol interfaces to the underlying link layers and physical networks are provided through the socket interface. This manages the multiplexing and demultiplexing of data messages and streams between user space processes and the physical medium(s).

3.3.3 Real Time Clock

At present DTN nodes require a reasonably accurate real time clock because the Bundle Protocol currently uses absolute value time stamps to set creation and bundle lifetime expiry points. It is also difficult to track bundles through a DTN without accurate time in each node it traverses.

3.3.4 Processor Power Management Controls

For DTN nodes that are intended to operate unattended for long periods without mains power supply, low power processors with power management capabilities are more or less essential. The platform used should provide access to the processor power management capabilities and provide support circuitry that allows these capabilities to be used effectively.

3.4 PERFORMANCE REQUIREMENTS

It is very difficult to specify highly precise performance requirements. The circumstances in which nodes will be operating will be very variable and weather conditions may play a large part in determining whether the node can operate for a sufficient portion of the day to satisfy the local users' communication desires.

3.4.1 Power Consumption

For nodes operating in locations where there is no mains electricity very low power consumption is essential. Values below 10 watts for the complete node electronics running at full power are desirable to minimize the size of battery and the area of solar panels required if operating unattended.

3.4.2 System Availability

For nodes operating unattended in areas where there is no main electricity, the combination of electronics, storage batteries and power collection mechanisms (photo-voltaics, wind generator, etc.) should aim to provide in excess of 90% availability of node functionality within an operating window of 16 hours a day, given average weather conditions for the season when and the location where the node will be operating.

3.4.3 Processor Performance

The main requirement on node processor performance is to achieve an adequate level of bundle exchange during communication opportunities. When using Wi-Fi connectivity the aim should be the ability to saturate the Wi-Fi link on a point-to-point basis when the Wi-Fi cards in use are providing maximum bandwidth. This should be tested with both very large bundles and a collection of small (minimum payload) bundles.

4. HARDWARE COMPONENTS

The following types of hardware will be in use in N4C:

- Animal tags – devices to be attached to reindeer running some wireless sensor network (WSN) protocol and/or limited N4C applications
- Hiker's PDA – ruggedized PDA running N4C applications
- Laptop – standard laptop (possibly ruggedized) running N4C applications
- N4C router board – an Intel Atom based board running N4C applications, possibly operating as an enclave gateway
- N4C Server/Gateway – a standard server box or VM, running N4C applications, but also typically well-connected to the Internet (unlike all of the above), though possibly connected to an emulation of a challenged network

4.1 ENCLOSURES

The N4C router board will be deployed in various enclosures:

- Semi-fixed base station – normally static, but luggable (transportable by helicopter), with photo-voltaic and/or wind generator power collection with high capacity storage batteries for use in Swedish Lapland summer trials and/or Kočevje. The unit is likely to be deployed in the open with no shelter so that the unit needs to be highly weatherproof (to IP65 standard or better) and capable of operating unattended for periods of weeks.

- Mobile data mule – an enclosure attached to a vehicle, which can be a car, helicopter, logging truck or skidoo; powered from the vehicle. In some cases the enclosure will be in shelter and need only be 'showerproof'. In other cases (e.g. on a skidoo) it may need to be fully waterproof.

4.2 NETWORKING

The following physical/link layers will certainly be used:

- Wi-Fi (IEEE 802.11a/b/g/n)
- 'SneakerNet' (transfer via mobile or solid state disks such as USB)
- Cellular telephone (GSM) connections
- Wired Ethernet

It is possible that the following additional physical/link layers may be used:

- Zigbee (IEEE 802.15.4)
- WiMAX (possibly modified)
- Nordic Mobile Telephone 450 MHz (NMT 450)[6]

[6] No longer generally relevant after 2008. The NMT 450 network which previously operated in Swedish Lapland was shut down after the first summer trials in 2008 and the system is no longer in use. However the technology may be used for local private connections.

5. SOFTWARE COMPONENTS

5.1 IMPORTED COMPONENTS

The following DTN infrastructure software components are the starting points for N4C. Note that more than one implementation of the architecture may be used:

- For the BPA, Storage Manager, DTN Routing, Convergence Layers, Communications Opportunity Management and Security Support sub-systems:
  - Relevant portions of the DTN2 reference implementation [DTN2], and
    - Prophet DTN implementation (omits Security Support) [ProphetDTN]
- For the LTP Convergence Layer, if required:
    - TCD's implementation of the LTP code base that is being implemented as a Convergence Layer module for DTN2

5.2 COMPONENTS DEVELOPED BY N4C

The following components are developed within N4C:

- Power Management sub-systems
- Remote monitoring mechanism
- Startup, shutdown and automatic configuration mechanisms to support unattended operation

It will also be necessary to improve and update some areas of the existing software. The following parts are known to need attention:

- A number of application proxies will need to be developed to support gateway functionality for some applications.
- Updates to the PRoPHET routing protocol both in the DTN2 and Prophet DTN implementations.
- Improving the logging system to ensure that bundles can be tracked through the network for analysis purposes.

6. PROCESS VIEW

The majority of the DTN infrastructure can be implemented as a single user level process. This process would use a number of threads to handle the various aspects:

- Communications Opportunity Advertisement
- Communications Opportunity Reception
- Convergence Layers (per communications link)
- Management interface
- Application server (API handler)
- Main event handler

Power Management and Remote Monitoring are handled in separate processes.

7. CONSTRAINTS

7.1 PHYSICAL IMPLEMENTATION

The DTN infrastructure is intended to operate on a range of platforms, but the low end is likely to be very constrained both in terms of physical size and resources. In particular it may be operating with only solid state memory. Depending on expected deployment, the equipment enclosures may require moderate to extreme weatherproofing.

7.2 POWER CONSUMPTION

N4C is targeting deployments where power is in short supply. As discussed in Section 3.4.1 the power consumption must be kept to a minimum.

7.3 UNATTENDED OPERATION

The infrastructure must be capable of operating unattended for long periods (weeks) and must not require the attention of a technically skilled local operator during normal operations.

7.4 NO KEYBOARD OR DISPLAY REQUIRED IN NORMAL USE

The infrastructure software must be capable of operating unattended in an environment that has a minimal local user interface. In particular it MUST NOT require the use of a display or keyboard in normal operation. Command and configuration input must be able to be provided by a remote terminal, initially over a CI connection or through downloaded control scripts. The eventual target would be to use DTN to configure and manage the system

PART 2: ARCHITECTURAL WORK CARRIED OUT IN N4C, EVALUATION OF ARCHITECTURAL DESIGN, AND RECOMMENDATIONS FOR FUTURE WORK

8. N4C WP2 OBJECTIVES AND EXPECTED RESULTS

The architectural work in N4C work in N4C was carried out primarily in the tasks of Work Package (WP) 2 with implementation of infrastructure components also being carried out in WP4.

The objectives of WP2 that are related to the architecture described in Part 1 of this document are described in the N4C Description of Work (DoW) [DoW] as follows:

- Provide an integrated architecture that will support seamless communications between applications in the communications challenged areas using the DTN paradigm and the existing network with a richer communications infrastructure using conventional IP technology.
- Research and develop techniques and protocols to support routing of IP packets and corresponding DTN bundles between the DTN environment and the IP environment, especially in situations where there are multiple possible interconnections between the regions.
- Research and develop techniques and software applications to provide an addressing mechanism for the DTN environment together with a mapping mechanism to dynamically link the IP addresses used by conventional IP protocols with the addresses used in the DTN environment.
- Research and develop security mechanisms to provide integrity, authentication and encryption of data passing through the DTN environment.
- Research and develop management applications to manage the addressing, routing, and security mechanisms developed under the previous objectives.

The DoW describes the expected results of WP2 as follows:

- Design of a network architecture that takes into account the various communications methods available in the neighbouring areas as well as the communications challenged areas
- Research on DTN friendly security measures.
- Research into routing in an opportunistic network and integration of this with routing in a traditional Internet.
- Continued research into routing in the DTN
- Production of scholarly papers on the requirements and solution of opportunistic networking.
- Production of viable protocol designs that can be introduced into the Internet Research Task Force Delay Tolerant Network Working Group (IRTF DTNRG) and submitted for future standardization.
- Criteria and guidance to System integration (WP7) and Real-life tests (WP8).
- Implications of the integration and test results (from WP7, WP8)

During the early stages of the project the consortium decided to use the Spiral Development Model so that the results of experiments carried out and documented by WP8 were fed back into both architectural design of the infrastructure software described here and the detailed designs of infrastructure software developed as part of WP4 and applications developed as part of WP3 and also in WP4. This process appears to have worked well and has resulted in a number of practical improvements to the equipment used in the later iterations of experimentation. Furthermore the long running experiments in both test beds, but especially in the Slovenian test bed where the experiments have been in progress continuously for almost the entire period of the N4C project (Summer 2008 through to April 2011), have provided a great deal of information about the practicalities of running a DTN infrastructure in realistic CCR environments as well as identifying problems that would probably not have come to light during shorter duration tests.

The practical work of creating the infrastructure and performing the experiments has in turn informed a considerable amount of theoretical work some of which has been already fed back into the later experiments. Much of this work has been presented to and documented for other researchers in the DTN arena through participation by N4C partner representatives in the IRTF DTN Research Group (DTNRG) [DTNRG]. The work that has been carried out for presentation in the DTNRG addresses several of the research aims of N4C WP2 and is documented in Section 9 together with work being undertaken in other standards organisations.

Other architectural aspects of the work carried out in N4C are reported in Sections 10 to 16 followed by overall conclusions and suggestions for future work items that have been identified as a result of the N4C project in Section 17. The implications and feedback for the experiments in the two test beds are considered for the architectural areas discussed.

Although not strictly architectural work, we have added two Annexes providing overviews of the system integration platform developed in WP7 and the various analysis tools that have been developed by the experimenters at LTU, MEIS, and TCD to present the data logged about the encounters between nodes and the exchanges of data during those encounters.

9. DTN RESEARCH GROUP AND OTHER STANDARDS WORK

9.1 BACKGROUND

Section 3.3 of the N4C Description of Work [DoW] states:

> "Partners in the N4C are already established as active participants in various international R&D environments and standards organizations. During N4C the partners will seek to increase the impact of the work by contributing to relevant conferences, working to integrate the N4C test bed into other R & D environments and working to standardize relevant aspects of the work. Impact will be achieved through presence in
>
> - Standards organizations and forums promoting interoperable solutions.
> - In particular N4C partners will continue to participate in the IRTF DTN research group, publishing Internet Drafts and seeking to have relevant work converted into standards in the IETF and published as RFCs (Requests For Comment)."

As presaged in the description of work N4C partners have concentrated their work on promoting interoperable solutions through the Internet Research Task Force (IRTF) DTN Research Group. The IRTF is closely associated with the premier international organization generating standards in the Internet arena, the Internet Engineering Task Force (IETF). As implied by the name, the remit of the IRTF is in areas where further research is considered to be required before full standardization can be undertaken. The mode of working of the IETF and IRTF in general and the DTN research group in particular is well matched to the 'research by experimentation' philosophy of the N4C project. As with N4C, the DTN research group supports and coordinates work that will allow diverse experiments and test beds to interoperate with the overall aim of demonstrating the value of DTN and improving its capabilities.

The output of the DTN research group includes well-reviewed documents categorized as either Experimental or Informational Requests for Comment (RFCs) published by the IETF that provide formal documentation of the protocols and techniques used in the flavour of DTN developed by contributors to the research group. These documents can be seen as pre-standards facilitating interoperation of multiple experimental deployments, but it is not intended or expected that they will necessarily be translated directly into standards: any standards that might evolve would be influenced by the outcomes of the experimental stage. This work is centred around the development of the Bundle Protocol [RFC5050] suite and the stewardship of the DTN2 Reference Implementation of the protocol suite.

9.2 CONTRIBUTIONS TO DTN RESEARCH GROUP

Partners Folly, LTU and TCD have all been actively involved in making contributions to the DTN Research Group during the course of the N4C project. Stephen Farrell of TCD is one of the co-chairs of the research group and has played an active role in ensuring that N4C's work in DTN is aligned with the aims of the DTN research group. In addition to Stephen Farrell, representatives from LTU and Folly have attended the four DTN research group face-to-face meetings either in person or via telepresence. Additionally, representatives of the partners have participated in subsidiary discussions especially related to management of DTNs outside of the main DTNRG.

During the development of the N4C test bed infrastructure and applications, N4C partners have investigated and researched various technical aspects of DTN protocols and documented this work through the publication of Internet Drafts for consideration and review by the DTN research group and the wider IETF/IRTF community.

Contributions were made to the following areas:

- Naming of DTN Endpoints through the dtn: URI scheme – documents
  - The DTN URI Scheme [URIscheme]
    - Adding the 'find' Operation to the dtn: URI Scheme [URIfind]
- Endpoint Discovery Protocol for DTN
    - The Delay Tolerant Networking Endpoint Discovery Protocol [EndPoint]
- Bundle Security Protocol [BSP]
  - Several updates have been made to this document to meet review comments both from within the DTNRG and from IESG and IRSG reviewers prior to its being accepted for publication as an Experimental RFC
    - Further discussion of the security work can be found in Section 10.

- PRoPHET Routing Protocol [PRoPHET]
    - Extensive experimental, simulation and theoretical work has been done on the PRoPHET Routing Protocol specification. There have been a number of updates to the Internet Draft that specifies the protocol. These versions have been presented to the DTNRG and review comments have been received from various member of the group. The document is expected to be submitted for formal approval very shortly. A summary of the work carried out by N4C partners in WP2 relating to the PRoPHET protocol is covered in Section 11; there is also a separate Appendix which explains the PRoPHET work in more detail.
- Self-Defining Numeric Values [SDNV]
- The Licklider Transmission Protocol [RFC5325], [RFC5326], [RFC5327], [LTPcl]
- DTN Management mechanisms. A sub-group of the DTNRG has been working on this topic intermittently during the period of N4C. Much of this work is still at a very experimental stage and it is not implemented in the DTN2 reference implementation as yet. N4C staff have been involved in this group. See Section 12 for discussion of N4C work in DTN management.
    - Drafts of MIB specifications have been published [BPMIB]
    - Diagnostic Interplanetary Network Gateway protocol intended for use with spacecraft [Ding]
    - A initial set of requirements for a protocol for remote management of DTN nodes has been published [DTNnetMgmt]
- Extending the bundle protocol to handle cases where nodes have no good clock [AltTime]
- Extending the bundle protocol to handle information-centric networking [QueryBlk]

The Internet-draft documents associated with a number of these areas have or are expected to become formally reviewed and approved RFCs in the near future. (There are six DTNRG documents currently in the RFC Editor's publication queue at the time of writing. See [RFCQueue].)

N4C staff have provided considerable support and expertise in progressing these and other DTN RG documents through the formal processes necessary for publication as RFCs during the course of N4C. In addition to providing a co-chair for DTNRG, Elwyn Davies (Folly) has acted as 'document shepherd' for the current set of DTRNG documents in the RFC editor queue.

9.2.1 Stewardship and Improvement of the DTN2 Reference Implementation

During the course of N4C responsibility for maintenance of the DTN2 Reference Implementation was transferred to staff from partners TCD and Folly. Alex McMahon of TCD has been primarily responsible for ensuring that this community resource was available to any interested party. The packages were moved to the Sourceforge repository using the Mercurial version control system [Mercurial]. A backup secondary repository was also maintained on Folly's server. The secondary server also delivers a number of additional pieces of N4C specific software which are being released to the community as Open Source software [N4Ccode].

In addition to basic maintenance of the DTN2 software, N4C staff have made a number of significant contributions to the package:

- Major additions to the logging system that ensure that bundles can be accurately tracked as they move through a DTN using DTN2 infrastructure. This work was triggered by the difficulties that were encountered in analysing the results of trials during Summer 2009. The improvements have significantly improved the value of the results and the ease of analysis for the Summer 2010 trials.
- Generation of a functional specification for the DTN2 software, incorporating a full manual of the user management commands that was not previously available.
- The LTP Convergence Layer has been integrated into the DTN2 code by staff from TCD.

9.3 SPACE STANDARDS

In addition to the DTNRG, the Consultative Committee on Space Data Standards (CCSDS) [CCSDS] has a DTN working group that is developing space standards for use in (mainly) future deep-space missions. That group have developed profiles of the BP and LTP suitable for use in space and N4C staff assisted in this process providing review and co-ordinating the work of the DTNRG and CCSDS groups. The Bundle Protocol has been tested for earth to spacecraft links during the N4C project period using the EPOXI (aka Deep Impact) comet exploration craft between its cometary exploration missions [EPOXI] and is currently being tested between the earth and the International Space Station [ISSDTN]. DTN is a serious contender for the infrastructure of space communications over the next decade.[DTNinSpace]

9.4 BROADER INTERNET STANDARDIZATION

As the DTNRG is part of the IRTF, which is associated with the IETF, N4C staffers have also been active in various roles in the broader Internet-standards work of the IETF, thanks in part to N4C, in particular as the DTNRG meetings frequently tend to be co-located with IETF meetings. In particular, Stephen Farrell has co-chaired an active IETF working group (DKIM) during the course of N4C and has recently been appointed as an IETF security area director, and is hence now a member of the Internet Engineering Steering Group (IESG). Elwyn Davies was until recently a member if the Internet Architecture Board (IAB) and continues to act as a member of the IETF's 'general area' reviewing team[7].

10. SECURITY CONSIDERATIONS IN N4C TRIALS

TCD and Intel Labs Europe deployed a delay tolerant network (DTN) in remote areas of the mountains of the Swedish Lapland for six weeks during the summer of 2010. The network provided web and email access for visitors and reindeer herders working 20-50km away from any power or communications infrastructure. While this was definitely a less threatening environment than faced by most Internet deployments, there were still a number of security issues that had to be addressed, and in fact, such a network deployment is useful to analyse since the security problem is more bounded and hence perhaps easier to understand. The security aspects of this deployment may also be useful for other researchers to consider when transitioning technology from the lab to the real world[8].

[7] This team supports the IETF Chair by providing 'generalist' reviews of all documents being considered for publication in the IETF document stream. The intention is to ensure that the documents are comprehensible outside the specialist area from which they originate as well as trying to pick up problems that might arise when combining new work with existing standards. We hope that the N4C work meets these criteria also!

[8] Note: Most of the text in Section 10, other than Sections 10.6 and 10.7, has been submitted for publication as a 'practical security' column for the IEEE Internet Computing magazine. It therefore contains some additional background which has been retained as it may in any case be useful

10.1 BACKGROUND

Delay- and Disruption-Tolerant Networking[9] (DTN [Fall08], [Farrell06]) is an active area of networking research on which TCD and Intel have been working for a number of years. Our emphasis has always been on attempting to test DTN technologies in the real world to the extent that our funding allows. DTN aims to provide Internet-like services for nodes that are in communications-challenged areas and has applications in space-networking, military tactical networking and sensor networks. Together with other partners in the FP7-funded Networking for Communications Challenged Communities (N4C) project[10] we have been deploying DTNs in Sweden and Slovenia over the past three years.

In summer 2010 we deployed our longest and most complex trial so far which involved five remote sites in the Swedish mountains the nearest of which was over 20km from any power or communications infrastructure with the furthest being 55km distant. There is generally no GSM service in this area[11] and satellite connectivity is both expensive and difficult due to the topography and latitude. The network in question consisted of Internet connected hosts in Dublin, connected via GSM to gateways that were located at two of the landing bases for the helicopters that are used by reindeer herders and hikers to get to the area serviced by our DTN. Figure 3 provides an abstract view of the network as deployed.

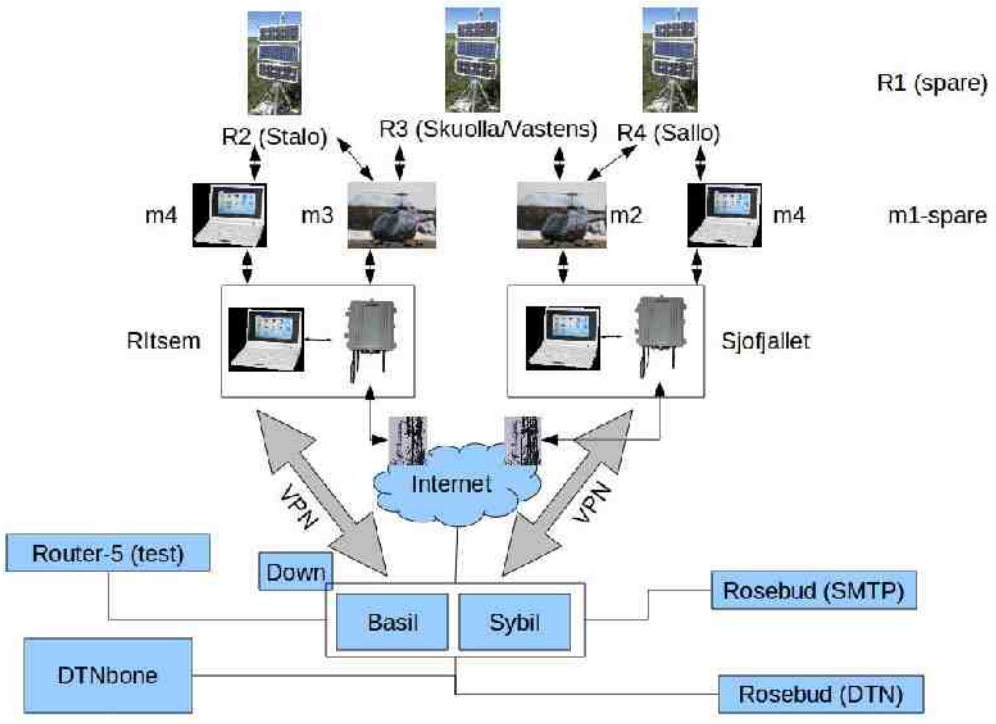

FIGURE 3 NETWORK DIAGRAM FOR SUMMER 2010 DEPLOYMENT

[9] http://www.dtnrg.org
[10] http://www.n4c.eu/
[11] During the trial, we discovered that climbing a mountain at one of our sites did result in getting enough GSM signal for a voice call. While we didn't use that for data in 2010 (it was a surprise!), we do plan to try a GSM relay on the mountain for our planned summer 2011 trial.

Two of the helicopters were equipped with single board computer (SBC) data-mules that store and forward messages between the Internet and the remote sites. We also used netbooks, handed to the pilots, as another kind of data-mule, since only two out of six helicopters had the SBC data-mules installed. At each of the remote sites, we deployed a solar-powered DTN-router/WiFi hotspot that we developed for the project. These act as WiFi hotspots and mail and web servers for the herders and tourists in the mountains. We used the Bundle Protocol (BP) - RFC 5050 [RFC5050] - for store-and-forward networking between all nodes. We provided netbooks and WiFi-enabled handhelds at the remote sites so that users could use a web browser and e-mail, either via a standard mail user agent such as Thunderbird [Thunderbird], or via webmail (using SquirrelMail [SquirrelMail]). Some users were able to use their own WiFi-enabled phones to use the service.

Web service consisted of either a 'URL fetch' service where the users entered a URL but then had to wait (usually overnight) for the corresponding content, or else 'pushed' web content where we updated content for a set of pre-configured web sites (mostly news and weather) each day. For email, users could setup new mail accounts via a browser after which they could use standard email services though of course each round-trip required packets to wait for two helicopter contacts. Each user's message store was replicated to each of the remote sites, as some users moved between the different locations over the course of the trials.

Figure 2 shows some of the contact graphs from our 2010 trial. In our 2009 trial, due to the various disruptions and queuing, the average one-way latency from the most remote site to the Internet was just over 15 hours. For additional details of our 2009 and 2010 trials see [DTNatTCD].

Even in this relatively unthreatening environment, there were a number of security considerations that needed to be tackled, both in terms of the network to be deployed and for the applications. We describe these from the 'bottom' up starting with physical security, then network access and ending with ethical approval.

10.2 PHYSICAL AND NETWORK SECURITY

Physical security was really only an issue at the gateway sites, where many people come and go, and where we had reports of a few cases where goods awaiting transport had previously gone missing. However, with the co-operation of the helicopter company, we were able to install our gateway equipment in a toolshed. This did however, at one of the sites, impact on the bandwidth available, as the Wi-Fi antenna was quite some distance from the usual helicopter landing site. It also similarly affected GSM bandwidth at the other site since the secure location was not ideal for GSM connectivity. For the remote sites, where equipment is vulnerable, we did not take any specific action other than to ensure that there was a person who would 'own' the equipment for the duration of the trial. The bulky nature of the equipment is generally a sufficient countermeasure. In the end, we had no issues with physical security at the remote sites, but one of our spare GSM devices did go missing whilst in transit between the gateway sites, for some unknown reason.

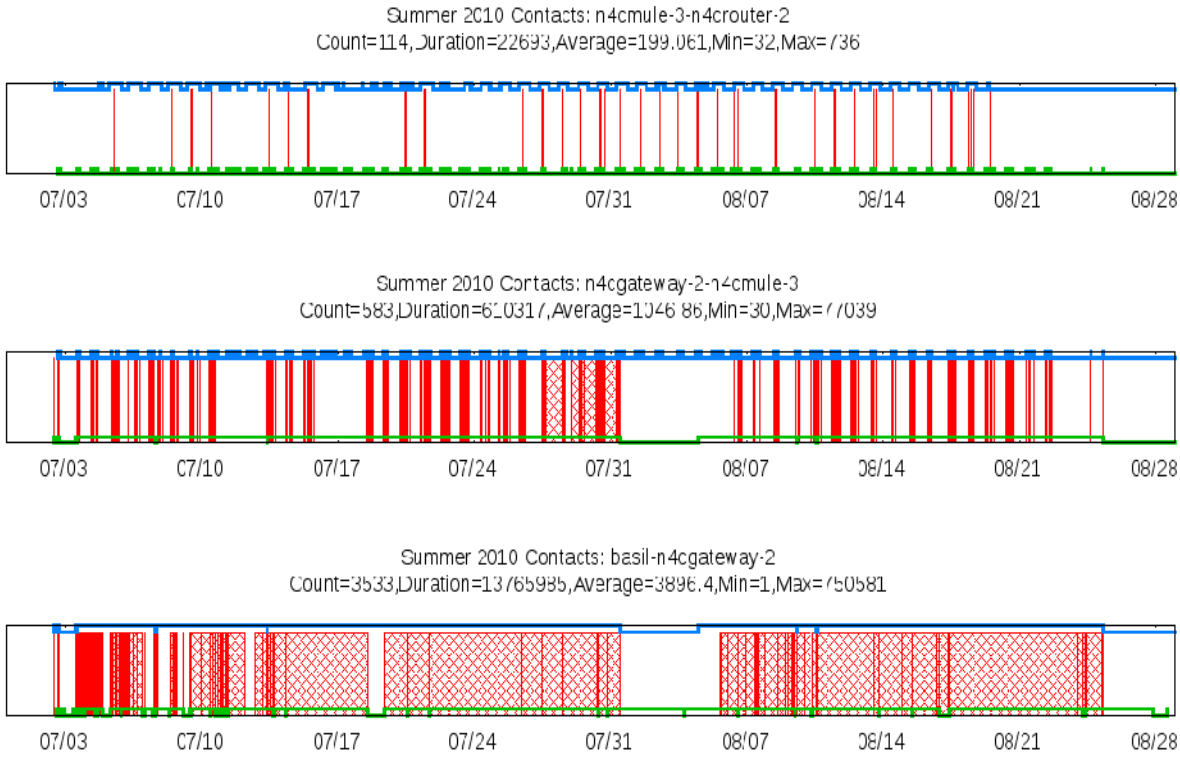

FIGURE 4 SOME CONTACT GRAPHS FROM THE 2010 TRIAL REPRESENTING CONNECTIVITY BETWEEN VARIOUS PAIRS OF HOSTS – THE SHADED AREAS REPRESENT TIMES WITH CONNECTIVITY. TOP: FROM MULE-3 TO A DTN-ROUTER/WIFI HOTSPOT ('ROUTER-2') IN ONE OF THE REMOTE SITES (STALOLUOKTA); MIDDLE: FROM GATEWAY-2 TO A HELICOPTER-BORNE DATA-MULE ('MULE-3'); BOTTOM: AN INTERNET-CONNECTED HOST ('BASIL') TO A DTN-GATEWAY ('GATEWAY-2') AT A HELICOPTER LANDING SITE. COUNTS REPRESENT THE NUMBER OF 'CONTACTS', OTHER FIGURES ARE IN SECONDS.

For network access, our DTN-gateways deployed at the helicopter landing pads were open Wi-Fi access points so that the helicopter borne data-mules could connect and send and receive data. On the GSM side, due to local restrictions on antenna location and distance from the cell-tower, these devices provided very low bandwidth (approximately 1-2kbps, with many disruptions as can be seen in Figure 4) so we were concerned that a user associating with the open access point could easily accidentally consume all of our bandwidth. Since the gateways were deployed for six weeks we decided to handle this via IP routing – when a node associated with the Wi-Fi access point at the gateway, its DHCP lease gave it a next-hop IP router that was firewalled so as to only allow BP traffic to be sent via TCP (port 4556). While this meant that a knowledgeable user could use our GSM bandwidth by simply re-configuring their IP stack, we felt this was sufficient for the threat level. We did consider turning on Wi-Fi security, however, that would have complicated our device configurations so we chose to leave all the Wi-Fi access points open. In the end, we didn't see anyone trying to use our GSM bandwidth so our approach seems to have been justified. (Practically, at the helicopter landing sites, everyone is concerned with getting on or off a helicopter and is not really interested in Internet access. They do, however, make GSM calls, which typically disrupted our data communications.)

Between the gateways and Dublin we used an OpenVPN IPsec tunnel in order to secure the traffic, but the main purpose of this was really to allow us to initiate connections from Dublin via GSM since

the gateway's Internet-facing IP address was dynamically allocated by the GSM network and the gateways were also behind NATs. Even though OpenVPN does not provide full IPsec (it uses SSL for key exchange), since all the links concerned were 'ours' this was acceptable.

We also made use of an SSH tunnel between our main Internet-connected host in Dublin and the public mail server that was used to send and receive email to our *village.n4c.eu* domain, which was used for all our mail accounts. Previous email trials had had problems with our mail being classified as spam due to the lack of an MX DNS resource record for the domain or because we sent SMTP traffic from 'unexpected' IP addresses, so we tunnelled all mail between Dublin and the existing mail host for the *n4c.eu* domain that had the appropriate DNS setup. In addition, our local firewall rules in Dublin would have prevented us from emitting SMTP traffic from the DTN host, so the tunnel allowed us to get mail to the right mail transfer agent for the *n4c.eu* domain. As far as we know our mail was delivered with no problems and inbound mail certainly arrived and was forwarded as required. The SSH tunnel was initiated from Dublin and again allowed us to initiate connections in the event of reboots. As with the Dublin-gateway tunnels, this was a case of using tunnels to achieve the end-to-end connectivity we required – in neither case did we have real confidentiality requirements, other than to ensure that we prevented any middleboxes from interfering with our traffic.

10.3 FUN WITH DNS AND LOGS

Due to the nature of our 'pushed' web content implementation, our DTN-routers in the village enclaves had to pretend to be the entire Internet so that browsers could access the 'pushed' content at that content's 'normal' URL. For example, we mirrored parts of the mobile edition of the Irish Times newspaper which we could access as http://m.irishtimes.com/ either from Dublin or while up the mountains in Sweden. This required our DTN-routers to effectively pretend to be the entire DNS to clients bound to the DTN-router's access point. This was done by installing a DNS server (BIND) on the DTN-routers with zone files for each top-level domain (TLD) that effectively just returned the DTN-router's IP address in response to any query. There are two security considerations arising – first, this scheme would not work were any client to implement DNSSEC which points to a future security challenge in deploying DTNs that service 'standard' clients who we expect will eventually start validating DNS responses using DNSSEC. We currently have no solution for this, but luckily for us our trial was finished prior to ubiquitous deployment of DNSSEC.

The second DNS issue related to our use of open access points – devices from at least one popular vendor of smartphones when connecting to an open access point attempt to 'call home' (via HTTP) to a well-known URL related to the operation of 'captive' web portals. While we did not investigate the exact nature of this exchange, it caused a problem for us. Since our DNS setup returned an A record (with our DTN-router's IP address) for the host in question, this caused the smartphone to request a web page we did not host. Our default behaviour in this case was to HTTP re-direct the client to an introductory page that described the trial. However, the smartphone interpreted this (in fact any HTTP 200 response apparently) as a captive web portal login page and so popped up a login dialog which of course was meaningless, and potentially dangerous should a user actually try to send some other credential of theirs in clear over the wireless network. To handle this, we simply setup our DNS to return a non-used IP address for this particular host, and when the client could not contact the host via HTTP/TCP it simply continued on to grant access to the open Wi-Fi network. Essentially, we had to black-hole this vendor's domain in order to allow its smartphones access to the open Wi-Fi!

During the trial we also discovered a number of other sites that also ought to have been black-holed, but were not, mainly to do with software updates. Of those, only one caused a real problem – in that case, the software update agent for a relatively popular netbook running a variant of the Linux Xandros distribution sent an occasional HTTP request to check update status, but when it got a response that was not in the right format, the agent simply re-issued the same request, at a rate of about one request every 8ms. This filled a filesystem with HTTP access log information on the DTN-router in question before our logrotate setup could move the logfiles to another partition with more capacity. In this case the solution in the field was to turn off the update agent, which luckily was running on the author's netbook and so was easily controlled.

Regarding logging, as an experimental deployment our configurations were set to produce extensive logging (resulting in about 12GB of compressed logs overall), however, after the trial, we did discover some cases where logrotate had been incorrectly configured, which unfortunately resulted in the loss of some logging information. Thankfully, most of that was duplicated either on other hosts or in other log files with different rotation periods, but we learned yet again that running the full system for an extended period in the lab is really required. The mis-configuration here was that logrotate was set to overwrite certain logs after 6 days instead of 6 weeks – due to our limited storage we needed logrotate setup so as never to fill the disk, so we always had to risk some loss of logging, but even though we had well-tested device configuration scripts that set the right values, during in-lab testing we had reset some of those (to 6 days) to test logrotate did the right thing, but unfortunately some devices were then deployed without being properly re-configured.

10.4 MAIL AND WEB SECURITY

For SSL for web and mail services in the remote areas, we simply created self-signed certificates on each device and so users who wanted to interact via SSL had to accept our self-signed certificate. This is not ideal, but was more convenient for us than getting real public key certificates for our devices which would further complicate device configuration.

For logging in to our DTN-routers and other hosts, we required SSH with public key authentication. For this, we simply generated a single key pair that could be used for any device and each person who needed login access had a copy of the relevant private key. For logins, the real concern was less with unknown users accessing the system, but with known users (mainly from our project-partners) who might otherwise attempt to 'fix' some problem, which in our experience typically causes some other worse problem. SSH was therefore really being used as a form of access control.

During the trial users were able to setup new mail accounts in the *village.n4c.eu* domain, and were then able to use those via webmail (using the *SquirrelMail* package) as well as from a traditional mail user agent (MUA) such as Thunderbird via IMAP and SMTP (either over SSL or in clear). In the main, we took this approach since we did not want to handle those user's credentials for other mail service providers. This lead to problems for users, with the main one being that they didn't have their address book and hence had problems remembering the mail addresses for people to whom they wanted to send mail from the mountains.

In order to setup mail accounts, users had to choose and enter a password. We also need to transmit this password, in order to create the user's account on each of the DTN-routers at the different sites, since message stores and mail delivery were synchronised across all sites. This required the development of a simple application layer confidentiality mechanism, since we expected that many users would choose passwords that they also used elsewhere, even though we told them that that was

a bad practice. We did this using a simple shared-secret that was installed on each of the DTN-routers, but so that the user's passwords were never stored or transmitted in clear. Of course, this meant that root access to the DTN-router would (with a relatively small amount of work) allow one access to those passwords, but in any case, since the mail servers also stored salted hashes of the passwords, this didn't really change the overall level of exposure.

Since we used the bundle protocol (BP) to handle all traffic in the DTN, we would have benefited here from the use of the cryptographic security mechanisms defined for that protocol, (BSP) [BSP] however our implementation of the BP doesn't yet fully support security so we did not have that option this time. Even so, we would still have required the application layer confidentiality mechanism in order to store the account information securely so there was again no real loss from the lack of bundle security.

Since users' mail was also replicated between nodes, however, the BSP would have been useful for protecting that traffic. The mail synchronisation in fact used a scheme of sending compressed differences between the Maildir format [Maildir] directories used by our message store, so even without encryption, a casual eavesdropper could trivially get to read users' messages from the synchronisation bundles. This however, is a part of the system where we could improve security in future.

Our 'URL fetching' application had one non-obvious security aspect, but that didn't really see live use so it's hard to tell if this is really needed or not. When submitting a URL for fetching, a user could mark the transaction as 'private', which indicated that only that user should see the results and should know about the existence of the transaction. Other transactions were 'public', so that anyone could see the cached results once they were returned. The idea was that in a small community, knowing that anyone had requested a certain URL (e.g. related to a medical condition) or the timing of a transaction could easily identify the person making the request. Our implementation of this feature was based on cookies – to make a private transaction the user had to accept a cookie which was required to view the results. This had the benefit of not requiring any new user account, but the downside that only users with their own browser environment could safely store the cookie. We hope to test this feature further in 2011 tests.

There were no real security considerations for the 'pushed' web content in the remote sites, however, back at the crawling site, a number of vulnerabilities arose with respect to the implementation of the crawling engine. The first, was that should a site know that we were going to crawl their content, then they could easily launch a Denial-of-Service (DoS) attack on our network, simply by linking to some very large object that we would then attempt to push to the remote sites. Secondly, as initially implemented, the crawling engine, which was a mixture of 'C' code and scripting, had a number of vulnerabilities. These could have allowed a malicious crawled site to subvert our crawling engine. For example, when crawling content, we were mirroring content into a local filesystem and had to include various sanity checks on the URLs crawled before we created files in our filesystem, for example, checking for 'foo/../../etc/passwd' type pathnames. In the end we didn't put too much effort into this, but a production environment, or a trial with more sensitive data might have to do much more work here.

10.5 ETHICS AND PERSONALLY IDENTIFYING INFORMATION

Lastly, there were a number of ethical issues that arise with trials such as ours. We are storing personally identifying information (PII) and so must take care with that data, controlling access to

those who need the data etc. In our case we published our policy so that it was accessible on each of the node's web sites, and brought to the user's attention before they created mail accounts. Essentially, we promised to take care with the data, not to publish any PII and to delete all the PII after a year. This means taking care with the data-sets we plan to publish for example listing the contacts between nodes and the other bundle protocol logs. All application layer information will be fully anonymized and only statistical information will be published. Since we are dealing with PII, we also now need to get ethical approval for such trials from a university committee set up for that purpose. Experimenters should plan for some additional documentation and delay that can be caused by such processes and should consider how they will anonymize and protect any PII that results from their experiments.

10.6 ACTUAL WORK VS. DESCRIPTION OF WORK

In the Description of Work (DoW) for N4C we stated that we would consider security and had planned to work on Authentication, Authorization and Accounting (AAA) for DTNs. The relevant section (1.1.3) from the DoW says:

> "The infrastructure of an opportunistically routed DTN of the kind envisaged here offers considerable potential for attacks on the integrity and authenticity of the data being carried. Relays will frequently be carried by persons who are not well known to the community and a malicious carrier would have ample opportunity to try and interfere with any traffic that they were carrying.
>
> An additional objective will be to ensure that, so far as is possible, any data carried through the DTN is protected against unauthorized tampering and any private data will be concealed from inspection by unauthorized entities. Research work is in progress on adding security capabilities to DTNs and the project will investigate appropriate deployments of this technology in the context of communications-challenged environments. The challenges of deploying security technology in the scenario envisaged are considerable due to the wireless infrastructure, 'store and forward' operation, the problems of key distribution, the expectation that many users will not be highly trusted, and the lack of sanctions that can be applied to ad hoc users.
>
> There are two additional security issues that require further research and development in order to meet the application and reliability requirements discussed above. Firstly, with DTNs being store-and-forward networks, and storage being a finite resource, there is sometimes a need to control access to that storage. In the Internet, this would be handled using an Authentication, Authorization and Accounting (AAA) protocol like RADIUS or Diameter. The project will therefore include work on a DTN AAA protocol, or more likely, a set of AAA extensions to the bundle protocol. Definition of an AAA model for DTNs, details on the approach to be followed:
>
> Some early investigations of this have been contributed to the IRTF DTNRG. These include a contribution by Hannes Tschofenig which was produced in the context of the Ambient Networks Project. The Ambient Networks Project is funded by the European Commission. The direction proposed included moving the information required for authorization closer to the user, into the local network, e.g. authorization based on nonfrequently changing attributes or an ability to regularly push revocation lists or access control information to the enforcement points. Issues of how the network authorization schemes will be bootstrapped will need further study. Secondly, DTN nodes are intrinsically highly vulnerable to denial-of-service (DoS) attack since they are generally 'exposed' nodes and cannot he 'hidden' behind a firewall. To make matters worse, a successful DoS attack on a DTN node can effectively 'kill' the node and seriously partition the network, since the typical recovery action (reboot) isn't easy with such scattered nodes. The project will therefore investigate additional ways in which DTN nodes generally, and nodes from the planned deployments can be made robust against DoS attacks."

We have done some of this but as can be seen from the above, we managed to tackle our actual problems in the trials mainly through the use of existing tools, and also didn't get to test the BSP due to the lack of a full implementation which would not have been practical to develop in N4C. BSP coding has been ongoing in the US since the start of the project, and so far, there has not been a suitable time at which we could take over that work, which remains incomplete – at present only the 'hop-by-hop' BAB ciphersuite from the BSP is part of the DTN2 code and for N4C we would really have been interested in the 'end-to-end' PIB and PCB ciphersuites.

Table 1 below isolates the parts of the DoW text that are actionable and describes what we did in each case, or why we did something other than what was originally planned.

DoW text	**What we did**
"Relays will frequently be carried by persons who are not well known to the community"	In fact, the relays we used were all known to the community (helicopters, pilots, N4C staff) so this threat did not arise in practice and we therefore did not implement countermeasures since the threat did not arise.
"ensure that, so far as is possible, any data carried through the DTN is protected against unauthorized tampering and any private data will be concealed from inspection by unauthorized entities"	We used IPsec and SSH tunnels for bundles transmitted over the Internet between the trial area and hosts in Dublin and the UK. Inside the trial area, we encrypted (mail credentials) at the application layer or obfuscated (message store synchronisation content) all data that was sensitive and considered this latter sufficient protection We therefore did not require the BSP in order to meet this requirement.
"The project will therefore include work on a DTN AAA protocol, or more likely, a set of AAA extensions to the bundle protocol"	Not done. See Section 10.7 below.
"The project will therefore investigate additional ways in which DTN nodes generally, and nodes from the planned deployments can be made robust against DoS attacks."	Our tunnelling setup, handling of DHCP leases at the gateways and security analysis of the web crawler were all areas where we considered the actual DoS vectors that could have affected our trials, and we consider that we met the requirement here. No DoS incidents were seen, other than the normal self-inflicted ones from any experiment.

10.7 DTN AAA

At the time of writing the DoW, we envisaged carrying out work on AAA issues in DTNs, and more specifically to investigate whether a Diameter [RFC3588] like protocol could be extended to handle AAA in DTNs such as those envisaged in N4C.

Since that time, we have begun, with others [Ahlgren10] to investigate information-centric networking (ICN) which is based on the concept of making information objects first-class objects in networking. The ICN approach for example would call for applications to be written so as to GET and PUT objects, regardless of their location(s) in the network topology and can leverage in-network storage in

order to reduce latency and congestion and improve availability. ICN has quite a lot in common with DTN and in the FP7 SAIL project [SAIL] we are involved in defining and developing a so-called 'Network of Information' (NetInf) where we again plan to visit Skuolla for DTN trials in 2011, and where the BP is one of the 'transport' protocols being proposed for use in 'developing region' ICN contexts. As part of SAIL, we have also proposed an extension to the BP [QueryBlk] intended to allow the BP to be more useful in an ICN context.

So how does this affect our view of AAA? We have come to the view that application layer security (e.g., confidentiality, data-origin authentication) may be better provided ultimately in the ICN, so that the information objects can be equally secure when accessed from a DTN as from a better-connected part of the Internet. (This doesn't mean the BSP [BSP] is useless, just that it serves a different purpose.)

In that context, we would like AAA handling, and specifically issues related to cache management, to be common regardless of whether the information objects are being accessed in a DTN or on the better-connected Internet.

In the SAIL project, we have partners who are major telecoms operators and suppliers and who will be working on these AAA aspects of ICN. Some SAIL partners also participated in the Ambient networks project referred to in the DoW text above. We therefore feel that that work is likely to produce an ultimately more interesting and more useful AAA solution, compared to the AAA work envisaged in the DoW.

We therefore re-directed our effort planned for DTN AAA to carry out some additional DTN trials involving DTN/WiMax that are reported in WP6.

10.8 SECURITY CONCLUSIONS

Even in such a non-threatening environment, deploying an experimental network required considering security and privacy aspects well in advance, implementing some security measures and handling PII for months after the end of the trial. Had our experimental network been either closer to a production environment or running in a more typical Internet context, then the amount of work involved would have increased substantially, for example, really hardening our web crawling system would require a lot of effort. Nonetheless, we ran our trial and didn't experience any security incidents that we know of, so our relatively minimalist approach seems to have been sufficient. We would hope that others deploying experimental networks and applications might be able to learn from what we've done, but most of all, that they would build security and privacy considerations into their planning – they will in any case, most likely be forced to do that in future as ethical approval becomes commonly required for web and networking experimentation and as can be seen from the above, the tools and mechanisms required are either freely available or else readily developed.

In terms of the security requirements posited in the project DoW, we consider we have met almost all of those, but using mechanisms other than those that were planned when the proposal text was written in 2007. The remaining area, AAA for DTN, was not tackled as explained above.

11. PROPHET

The networking use cases that N4C set out to address in the test bed CCRs included using user-carried devices as data-mules in a paradigm that we called 'the user is the network'. This scenario was originally developed during the predecessor SNC project. Such a 'network' would inevitably not have

a stable, well-known topology such as characterises the CI or even the degree of semi-scheduled regularity that occurs with the helicopter data-mules that provided most of the data carriage between the CI and the village routers used in the experiments described in Section 10.1.

In this use case data bundles are exchanged when users (and their carrier devices) have encounters. Since human activity is involved, these encounters are not random in the sense of Brownian motion with 'white noise' type statistics. Rather the patterns of human activity will be imposed on the frequency of encounters and the identities of the encountering nodes.

If the activity was really described by Brownian motion then the best message routing mechanism to use would (probably) be 'Epidemic' routing [Epidemic]. This involves a pair of encountering nodes coming away from the encounter with each carrying the union of the sets of bundles that the two nodes had before the encounter. Then assuming that the lifetime of the bundles is sufficiently long and there is enough storage in all of the nodes the bundle will reach its destination in due course. The set of paths that the bundle explores in its 'search' for the destination can be seen as a *delivery path tree* rooted at the source node for the bundle. On the positive side the successful delivery path used will be the optimum one since the bundle explores *every* possible path; on the negative side the system is profligate in its use of both storage and bundle exchange capability (length of communication opportunity during an encounter multiplied by the available bandwidth for bundle exchange).

To improve on the resource usage of the Epidemic routing solution, Avri Doria and Anders Lindgren, then both at LTU, developed the novel routing protocol called the Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) during the SNC project. This protocol aims to 'prune' the delivery path tree that can be used to describe the various paths through the network taken by bundles to eliminates paths that are relatively unlikely to lead to delivery of the bundle from the set explored by the bundle.

PRoPHET belongs to a class of routing protocols that are described as *dynamic*. This is because they are applicable to networks such as the kinds that N4C was envisaging where there is not a fixed *static* topology, but rather the nodes interact dynamically.

The protocol was defined and the specification published as an Internet Draft during SNC, and had just been adopted as a research group work item at the beginning of N4C. An implementation of PRoPHET and the Bundle Protocol was developed for SNC chiefly by Samo Grašič, subsequently a PhD candidate working on N4C at LTU. This DTN infrastructure software matches the architecture defined in Part 1 of this document but it currently uses an earlier version of the Bundle Protocol [RFC5050]than the DTN2 Reference Implementation [DTN2] and so is not directly interoperable. This infrastructure is known as **Prophet** (to distinguish it from the routing protocol PRoPHET) and was used for experiments during N4C in both test beds.

11.1 OUTLINE OF PROPHET ROUTING SCHEME

Effectively PRoPHET dynamically builds an internal model of the statistics of the encounter pattern in the DTN network of nodes that it encounters. The model is in the form of a set of *delivery predictabilities* (DPs), a single probability value for each node that it has directly or indirectly encountered in the reasonably recent past. To determine if a bundle should be passed to an encountered node, the DPs for the destination of the bundle in the two nodes are compared. If the DP in the node that doesn't currently have a copy of the bundle is greater than or equal to the DP in the node that does have the bundle, then it is a candidate for exchange.

Using the actual encounter events and the times between them (the 'History of Encounters' in the title) plus exchanges of DPs for other nodes during encounters ('Transitivity' – if node B meets node C frequently then it is a good idea for node A to pass bundles destined for C to B if A meets B frequently), the protocol evolves the DPs in a way that should capture the essence of the encounter pattern. Thus PRoPHET prunes the delivery path tree so that bundles are not offered to nodes that don't offer a better chance of a set of encounters leading to delivery than the current node. Nodes that are offered bundles are not obliged to accept them and can use additional criteria to determine if accepting the bundle would be a good use of resources. For example bundles with longer remaining lifetimes might be preferred over those with shorter as the bundle would be more likely to expire before delivery, having taken up storage that could have been better used. For more details of the operation see [PRoPHET]. More details of this work are in Annex 3 of this deliverable and also in a paper being submitted to the CHANTS 2011 workshop.

11.2 USE OF PROPHET IN N4C

The Prophet infrastructure package has been central to two sets of experiments during N4C:

- The environmental monitoring experiments carried out by MEIS with the test bed in the Kočevje region of Slovenia, and
- The general communication and monitoring experiments carried out by LTU with the test bed in the Swedish Lapland test bed area.

In addition some further experiments were carried out to link the two test beds during the final stages of experimentation.

The Slovenian test bed has been in almost continuous operation since the first Summer tests of N4C in 2008 and has produced excellent results, particularly about the long term reliability of the software and the behaviour of the PRoPHET protocol during such long term tests. Long term testing during N4C has proved to be extremely valuable in highlighting situations that could cause problems with a future permanent deployment and we have been able to make a number of improvements to the equipment and software as a result of this dedicated usage.

The Swedish Lapland test bed was more intermittent in operation but a small number of nodes were also operated for extensive periods at LTU to exercise the software. The Swedish Lapland test bed also provided more challenging weather and power availability conditions especially in the final batch of Winter Testing in 2011 [SwdWinter2011].

The two test beds were also interconnected during the final stages of testing in winter 2010-11.

A set of reports detailing what was done during the six testing periods of N4C are available as milestone reports for WP8. An overview of the Slovenian test bed is also available in Deliverable 2.3 which makes an offer to integrate the Slovenian test bed into the FIRE federated test bed network. N4C partner MEIS are intending to keep the Slovenian test bed in continuing operation and use it for other experiments.

The Prophet software has gone through a number of improvements as a result of this testing in line with the spiral development model. Many of these are practical issues that do not affect the architectural design, but the need for sophisticated power management and storage resource management have been emphasised by the trials.

The long term trials have also picked up some unusual situations in which the PRoPHET DP evolution algorithms do not result in the DPs providing a good model of the encounter patter as will be described in Section 11.4. Improved versions of the algorithms have now been identified through simulation and theoretical work, and solutions that were identified in time have been exercised in later experimental periods.

Practical experience has also highlighted some significant issues with regard to how nodes are able to discover that an encounter could occur. This will be discussed in Section 16.

We have also gathered information about the intervals between encounters which will allow us to better select the parameters used in PRoPHET to control the DP evolution. These statistics also highlight the distinction between the short intervals that result from artefacts of the Wi-Fi communication link and the much longer intervals that represent the underlying behaviour pattern. Providing a simple way for PRoPHET to separate these phenomena has been a major step achieved during the latter stages of N4C. See Section 11.5.

11.3 SIMULATION VS. EXPERIMENTAL EVALUATION

N4C was specifically set up to perform research by experiment. In the field of dynamic routing protocols for DTNs this has proved to be highly significant. As reported in our DTN State of the Art document [DTNstateArt], a very large number of dynamic routing protocols designed for DTNs have been proposed. Of these very few have actually been exercised in a practical experiment and, even today, after several years of work, PRoPHET is the only protocol which has a full protocol specification

The remaining 'protocols' for opportunistic networks have been explored only with simulations. Furthermore the models that have been used for the simulations are very often purely random encounter models (the Brownian motion scenario). These models are highly unrealistic and lead to inappropriate conclusions.

Early on during the project we became aware of a paper introducing the RAPID dynamic routing protocol [RAPID] and comparing this novel proposal with several others including PRoPHET. In this paper PRoPHET showed up very poorly. Samo Grašič set about investigating why this should be. In the course of trying to validate these results, he discovered that the PRoPHET implementation that had been used was flawed and did not match the specification, thus invalidating the RAPID comparison.

However in order to provide a more positive outcome, rather than just casting doubts on the results, it was decided to build a set of simulations in the One Simulator [One] that is a well-studied and independently verified system. It also provided a set of more realistic models, including the 'working day' model that uses probabilistic patterns similar to the sort of human interaction mobility pattern that we believe PRoPHET is well suited to. The PRoPHET implementation in the One Simulator has now been used for a number of experiments and has proved useful in validating several incremental improvements to the PRoPHET protocol, during N4C. The updated version of PRoPHET has been shown to produce better delivery percentages and better storage resource utilisation than older versions and some other protocols. A paper, intended for the CHANTS 2011 workshop, is in preparation documenting the results.

The simulation work taken together with the experimental usage of PRoPHET led the protocol authors to rethink various aspects of the PRoPHET protocol as documented in Sections 11.4 and

11.5. This work ultimately lead to the creation of a new version of the PRoPHET protocol that is described in Section 11.6.

The One Simulator is generally used to provide a number of measures of effectiveness of a (dynamic) routing protocol that have become conventional in this field of research. The primary figure of merit would usually be the percentage delivery for bundles often combined with the delivery latency (how long it takes to deliver a bundle). For a general comparison of protocols, these are useful measures, but for PRoPHET there is an intermediate measure which requires monitoring the effectiveness with which the DPs model the mobility pattern. Elwyn Davies developed a simple spreadsheet based simulator that allowed the patterns of DPs that result from various encounter patterns to be visualized and assessed. One area of further work that is needed is a quantitative way of assessing the output of this simulator to give some figures of merit when comparing algorithms. At present the evaluations have to be made by inspection and require a value judgement from the user.

Thus overall N4C has seen an effective combination of experimentation, simulation and theoretical work driving several new releases of the Prophet infrastructure software and the PRoPHET specification. The final piece of the process is that the connection logs from the experiments in Summer 2010 are being converted to a form where they can be used to drive simulations in the One simulator. This was an important aim of the project as it gives researchers another set of real world data that can be used to check on the effectiveness of changes to the PRoPHET routing protocol and, indeed, any other dynamic routing protocol that is targeted at the sort of opportunistic encounter scenario presented by the N4C test bed.

11.4 UNTANGLING THE CHAIN

The testing performed by MEIS during the Summer 2009 period [SlvSummer2009] identified a particular situation where PRoPHET was not behaving as expected resulting in bundles becoming 'stuck' at a node and not being forwarded to its final destination. The situation was identified as a case where a bundle had to pass through a long chain of nodes to reach its destination. Some of the nodes in the early part of the chain encountered each other repeatedly but only encountered the appropriate node for forwarding the bundle to its destination les frequently.

Samo Grašič analysed the situation and generated a simulation that was able to reproduce the unexpected behaviour during the autumn of 2009. It was clear from this that in the particular circumstances the evolution equation that was used to implement the transitive property was leading to inappropriate changes in the DPs. The original authors of PRoPHET together with Samo and Elwyn Davies looked again at the evolution algorithms and eventually came up with an improved version of the transitivity equation. Using a different form of simulation Elwyn Davies was able to show that this algorithm was much better at modelling the mobility pattern than the original version.

The changes to the algorithm were simple to implement and the improved algorithm was used with considerable success during the trials in Summer 2010 and afterwards.

11.5 WI-FI AND PARKING LOTS

It has been theorized that, in its initial form, PRoPHET would be likely to produce inappropriate evolution in its DPs if a group of nodes came together and repeatedly exchanged their sets of DPs. This was named the 'Parking Lot Problem' as the situation that was discussed involved a number of users with DTN devices congregating in the (Jokkmokk) supermarket car park or 'parking lot' to use

American terminology. During most of our experiments we have not had enough nodes in the same place at the same time for this situation to be a problem.

However on analysing the data from experiments in Summer 2010 it became clear that even without clusters of devices coming together we were seeing large numbers of encounters that were separated by only a short time – the human activity patterns would normally be expected to show encounters hours or days apart for the large part as shown in Figure 5.

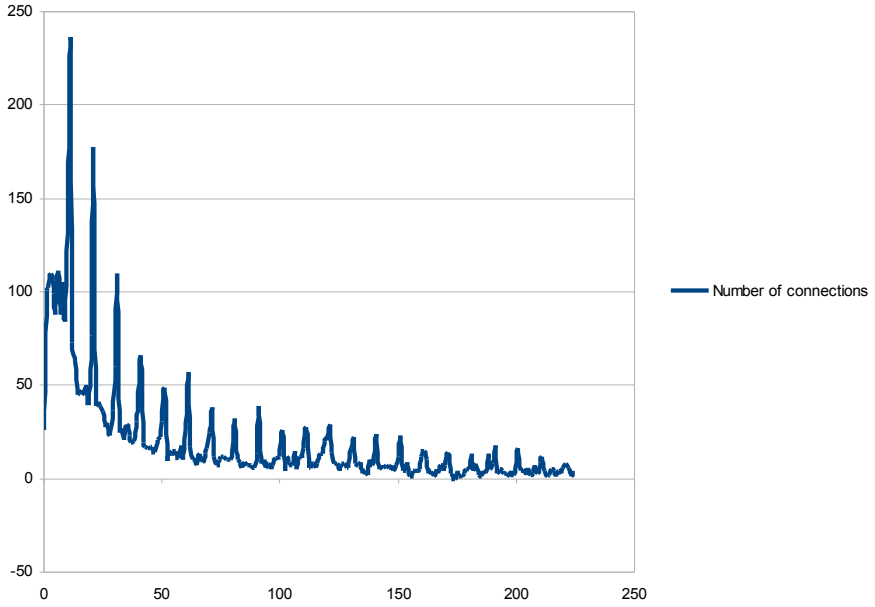

FIGURE 5 INTERVALS BETWEEN CONNECTIONS

The quasi-periodic nature of this plot is related to the time between discovery beacons used by Prophet nodes to signal their presence to other nodes. Many of these connections are actually reconnections due to the Wi-Fi link being lost and restored, and counting each of them as a new encounter between nodes exaggerates the number of human encounters. If PRoPHET treats these short period reconnections as new encounters then the DP for the encountering nodes will be increased too much and the model of the mobility pattern will be distorted. This is very similar to the theoretically postulated parking lot problem.

Having identified this problem, we endeavoured to find a way to avoid this distortion with a mechanism that would not require additional information exchange and would also not require large amounts of extra information to be stored in each node. After one false start, Elwyn Davies came up with a solution that satisfies these requirements. The solution has been tried out in small scale simulations and is being tried in larger simulations. In combination with the improvement to the transitivity evolution it appears that it also resolves the parking lot problem. It was found too late on in the project to be used in N4C experiments but is expected to be deployed in the Slovenian test bed later in 2011.

11.6 PROPHET VERSION 2

The Internet Draft specifying the PRoPHET protocol had just been accepted as a DTN research group working item at the beginning of N4C. This decision signifies recognition that the work on

PRoPHET is thought useful and significant for the overall development of DTN. In the course of N4C, Samo Grašič and Elwyn Davies became co-authors of the draft with the originators of the concept. Elwyn Davies has acted as editor for a number of revisions of the draft that have been published during N4C.

The original version of the PRoPHET protocol was designed before the Bundle Protocol was completed in 2007. At the start of N4C, the version of PRoPHET did not reflect some of the changes that had been made to the details of the Bundle Protocol, notably the importance of bundle fragments and the components of the unique bundle identifier. It was also decided that the published specification should concentrate on using a reliable, connection oriented protocol such as TCP for the PRoPHET data exchanges. As originally envisaged PRoPHET was designed to run over almost any transport including UDP, and even directly over IP. Delegating reliability to the transport protocol simplifies the PRoPHET state machine somewhat and using a connection oriented protocol allows implementers to design more easily for nodes with a large set of DPs and/or lots of bundles to offer; it allows the information to be packaged into several packets before being submitted to PRoPHET rather than giving the complete set to PRoPHET as one large memory buffer and having PRoPHET fragment the data for transmission.

Accordingly a number of improvements have been made to the specification, including

- the improved transitivity algorithm discussed in Section 11.4,
- the improved encounter management discussed in Section 11.5,
- additions to the bundle identification to bring it into line with the most recent Bundle Protocol,
- additions to allow bundle fragments to be offered for exchange,
- specifying TCP as the transport protocol to use (although any other reliable, in order, connection oriented protocol would work equivalently, such as Bluetooth RFCOMM),
- tightening up the specification of the protocol state machine, and
- providing much more comprehensive advice on setting the various configuration parameters used by the DP evolution algorithms. Much of this advice relates to understanding the characteristic timescales of the mobility pattern. This work is discussed further in Section 11.7.

The new specification for PRoPHET Version 2 is now published as version 9 of the Internet Draft. It is currently in the approvals process for release as an Experimental RFC.

11.7 THE IMPORTANCE OF TIME

It has become steadily clearer during N4C that *time* is significantly more important to DTNs than to the CI. Clearly DTNs need to take time into account because they are trying to handle delays much greater than the 'speed of light delays plus a bit' that tend to characterize the CI. Indeed, in the CI, for the most part developers and researchers have tried to minimize the role of time. 'Always on' networks and relatively uncongested networks have mostly reduced network delays as seen by applications and human users to near insignificance by human standards. Indeed all too often applications and protocols treat long delays as an error, and in many cases the problem is reported to the user as if it was the user's fault that the network cannot respond in time. From the point of the user experience this will not do for a delay tolerant network, and DTN applications need to explain delays and work with them to make the application useful even so. From an operational and psychological point of view this is important but in this section we are much more interested in how time affects the behaviour of the network: for the CI we try to disguise the passing of time but that is not possible in DTNs.

There are a number of areas where time has to be taken into much greater account in a DTN in addition to just allowing for delay. Some of the most important are:

- **Nodes are 'on their own' for much of the time**: consequently bundles are also on their own for much of the time. There has been a lot of argument in DTN research circles about the decision to embed absolute times into bundles, primarily to determine the expiry time of the bundle. This requires a reasonably well-synchronized wall clock device in each node. This can be a problem for very low end devices such as stand-alone sensors. Researchers from TCD have been investigating this issue further and have published an Internet Draft showing some possibilities for using relative times rather than absolute times at least for some nodes [AltTime].

- **In a network where PRoPHET is applicable, mobility patterns will have a characteristic time period**: To correctly set up the parameters of PRoPHET it is necessary to have some insight into the mobility pattern. This will involve time intervals such as the expected inter-encounter period (ignoring the sort of pseudo-encounter caused by Wi-Fi artefacts as described in Section 11.5) and the length of time taken to deliver a bundle. With a decent estimation of these times the protocol parameters can be set up in a logical way. Suggestions for doing this are included in the PRoPHET Version 2 specification. Future work might allow the parameters to be modified dynamically if the mobility pattern alters. By monitoring the interconnection times and recording how bundles pass through the network it might be possible to optimize the protocol during operation (subject always to stability concerns). Some earlier work looking at how time (scales) affect the appropriate choice of parameters was carried out by Jouni Karvo and Jörg Ott and presented to CHANTS 2008 [Karvo08]. This work looked at how to set up PRoPHET in a highly mathematical way; it was considering PRoPHET version 1 and did not look closely at the transitive evolution algorithm

- **The length of communication opportunities**: determining which bundles to exchange with an encountered node is a multi-dimensional resource optimisation procedure, as was recognized by RAPID. The length of time that nodes are in contact and the effective bandwidth between nodes during the contact are key factors in the process.

- **Power management**: In a CCR there may not be enough power available for a node to be 'always on'. Frequently the node will need to go to sleep or even shutdown completely for a proportion of the time, perhaps on a regular schedule if it is known in advance that usage is low at certain times. Power management is considered further in Section 12. The difficult topic of symmetric discovery (see Section 16) is also relevant to this concern.

A paper discussing the time issue in more depth is in preparation. It is hoped that it will be accepted for CHANTS 2011.

11.8 PROPHET CONCLUSIONS

Overall PRoPHET routing and the Prophet infrastructure software have been successfully and widely used during N4C. The aims of N4C as regards PRoPHET have been largely carried out in WP2 by demonstrating and improving PRoPHET.

One area where we were not as successful in using PRoPHET as had been hoped relates to problems that were encountered with battery lifetimes and *ad hoc* Wi-Fi connectivity between mobile phones and Internet tablets. Our intention had been to make a more extensive demonstration of the 'user is the network' paradigm using PRoPHET and this sort of device, but the issues with hardware and also

with the PRoPHET implementation in DTN2 severely limited real world experimentation. However the work that was done has served to highlight what we consider to be an extremely important architectural problem which deserves attention in future work. This matter is discussed further in Section 16.

The work carried out in N4C has significantly improved the PRoPHET specification, and the long lived experiments have both demonstrated the effectiveness of PRoPHET and shown up areas where improvement was needed. The areas that are directly relevant to the architectural work carried out in WP2 are summarized here; various other improvements have been carried out as documented in the WP8 deliverables and in Sections 12 to 15 below.

We have also achieved the aim of capturing some real world connectivity patterns that will be helpful for further research work using simulations.

12. INTEGRATION WITH THE LEGACY INTERNET

As stated in the DoW (Section 1.2):

> A major goal that will be addressed by N4C is the integration of the DTN portion of the network with the legacy Internet. This will be crucial in order to ensure that the N4C test bed can in future become a part of other future Internet test beds, which, for a network in a challenged environment is non-trivial.

The test beds in N4C have been successfully integrated with the Legacy Internet (i.e., the CI). A number of the applications deployed have demonstrated how data can segue from the CI to the DTN and back again.

The Summer tests in 2010 in Swedish Lapland also employed two gateways to provide redundancy and potentially a better delivery path for some bundles.

In practice the integration was mostly achieved by using application level gateways and proxies rather than direct packet - bundle conversion at the lower layers of the stack. These gateways were able to handle the situation of replicated bundles referred to in the DoW.

For the purposes of our experiments, with the volume of data being transferred and the restricted number of DTN nodes involved, we decided that devoting resources to optimizing the gateway to be used as the ingress point to the CCR, as suggested in the DoW, was inappropriate. The architecture would readily support such an optimization in the future and this work should be continued.

13. POWER MANAGEMENT

The consortium was aware at the outset of the N4C project that power management would be a major concern for DTN nodes running in CCRs. As stated in the DoW (Section 1.2):

> "In order to power a number of the autonomous devices envisaged by this project we need to develop novel strategies to maintain their activity during extended periods."

In practice the experimental teams have all learnt that we still probably underestimated the challenge of running unattended equipment for long periods in areas where there is no mains electricity and the environment for harvesting energy locally is also challenging.

In the event designing in, developing and deploying a power management sub-system in many of our nodes has taken a good deal of resource and required considerable thought.

The spiral development model has again served us well in this area. All the experiments have benefitted from at least two iterations of design, deployment, experiment and analysis.

The village router designed by TCD and Intel [NodeDesign] has benefitted particularly from introduction of Intel's latest Atom range of low power processors. These processors have an extensive suite of power management capabilities and N4C were lucky that the availability of suitable single board computers (SBCs) using the most recent chips coincided with the needs of the 2009 and 2010 experimental cycles. In practice the boards for the 2009 trials did not have a BIOS (built in platform control kernel) that allowed full access to the chip capabilities. This was remedied in time for the 2010 iteration and full advantage was taken of the improvements.

The Slovenian experiments also used Intel Atom based boards but of a rather earlier release. In all cases the basic SBC consumed around 8 watts of power when fully operational.

The details of the power management work for the 2010 version of the TCD/Intel village router are described in a paper that has been submitted for journal publication [SolarPwrMgt].

The power management work for the Slovenian test bed is described in the various milestone reports describing the tests.

As mentioned in Section 11.7, controlling when the nodes are accessible is one key aspect of managing the power consumption of the equipment and matching it to the available power stored in the unit.

It has also become abundantly clear that CCRs, especially in the Arctic region, are extremely challenging physically for the equipment. For example, a wind generator deployed in the final set of trials in the Swedish Arctic during February 2011 destroyed itself and damaged several other pieces of equipment [SwdWinter2011]. Even if the equipment coped with the weather, there were periods when the sun was hidden for extended periods making it difficult to maintain battery charge. Over specifying (for example) the solar photo-voltaic collectors might be a way of coping with poor weather but it is clearly important to know the correct compromises to maintain an adequate service without making the equipment too expensive and overly bulky, as it needs to be transported from time to time.

The work on power management proved to be significantly more challenging than we anticipated and the results obtained have been very useful.

14. STORAGE MANAGEMENT

The long term experiments carried out in both Slovenia and Sweden brought out a number of practical and design issues for the storage management on the equipment that we used.

In both cases the larger scale and longer duration of the experiments in Summer 2010 involved much larger number and volumes of bundles. This has highlighted some weaknesses in the permanent storage mechanisms.

The Slovenian experiments use equipment with mainly solid state (Compact Flash or CF cards) to hold the permanent storage. In general variants of the Linux operating system were used for these nodes. Experience has shown that there are a number of problems with using CF cards, some of

which are common to most Linux filing systems and some of which seem to be specific to CF systems:

- Single directories with very large numbers of files become very slow to access both for read and write. There are well known solutions to this problem but both Prophet and DTN2 were not written with large production systems in mind and store all bundles in a single directory. If even larger scale trials are envisaged then it will be necessary to rewrite the storage systems to use a hierarchical system. The Linux filing systems perform much better with the files distributed in multiple directories in a tree structure.
- CF filing systems become very slow when nearly full and are generally quite slow when writing large numbers of small files due to the need to update the index structures. This caused problems when nodes were trying to exchange both large number of small bundles and very large bundles. If the communication opportunities were short it seriously limited the number of bundles that could be exchanged[12].
- As far as we are aware we did not manage to run into the write cycles limit on any of the CF cards used, but that could be a consideration in future long term usage.

Careful handling of log files is also essential. Particularly when experimenting in the test beds log files can become very large and both DTN2 and Prophet are not very robust in the face of a full filing system for the log files.

15. DTN MANAGEMENT AND LOGGING

At the outset of the project N4C partners were aware of and had participated in some research into management techniques for DTN nodes. Researchers at NASA are working with DTN as a future infrastructure for communication with spacecraft and management control and status reporting will be a key requirement for operational deployment of DTN in space.

Until recently the standard management technique for CI nodes promoted by the IETF has been to instrument protocols and components with Management Information Bases (MIB) and use the Simple Network Management Protocol (SNMP) to transfer information between client MIBs and management stations. Recently an alternative scheme has been developed call *netconf*. In the early stages of N4C we investigated whether it would be possible to adapt the netconf mechanism to the very different environment of DTN. It became clear that N4C did not have sufficient resources to both to develop the specification and implementation of the management information representation in netconf, given that there is still little prior art available in this area, and to implement a suitable, reasonably secure transport system.

NASA has preferred to remain with the 'traditional' MIB scheme and has continued work in this direction. N4C researchers have been tracking this work but it has progressed relatively slowly and it is only now that a firm specification is emerging.

Consequently the management and logging applications for the test bed experiments have relied on the existing control and status interfaces available in DTN2 and Prophet. In both cases status information has been extracted from the infrastructure by a separate application and then the

[12] The cards being used were 'standard' CF type. It is not definitely known if the solid state disk modules that can plugged in disk interfaces such as Parallel ATA or the more modern CFast drives suffer from the same problems – we suspect not. Early trials used 'industrial grade' CF cards which are faster and support more write cycles, but the SBC in use did not support the higher power consumption of these devices.

information has been parcelled up into a bundle message and sent to the management station over the DTN in addition to being stored locally.

One outcome of the first large scale experiments in Summer 2009 was learning that in order to effectively analyse the behaviour of the DTN, it was essential to be able to accurately track the path a bundle takes through the network and identify any events associated with the bundle in the nodes it passes through. Up to this time, the logging mechanism in use was that already built into the infrastructure software. The logging output was primarily designed to help developers debug the software rather than track bundles. Tracking bundles where the DTN2 infrastructure was used proved to be particularly challenging. It was also not helped by at least one node with a clock that was set to the incorrect time.

TCD determined that some additional logging at the *info* level was needed to make bundle tracking much easier. This was implemented prior to the Summer 2010 trials and is now a standard part of the DTN2 software. Tracking of bundles was much more effective in the Summer 2010 trials.

16. SYMMETRIC DISCOVERY

As has been reported in the N4C report for Period 2 and in our analysis of the application work [N4CappEval], our attempts to use Internet tablets such as the Nokia N810 and mobile phones including the Nokia N900 as combined data mules and user clients for DTN has been severely hampered by hardware constraints.

It was anticipated that it would be possible to use the *ad hoc* mode of Wi-Fi to allow pairs of these nodes to exchange data during encounters. However problems with battery lifetime when using *ad hoc* Wi-Fi and, to a lesser extent, incompatibility between implementations of *ad hoc* Wi-Fi from different manufacturers meant that it was not possible to make useful long term tests with unskilled users.

It appeared that a possible solution had appeared in the form of Low Energy Bluetooth which seemed to offer the chance to provide regular beaconing with very low energy costs – this is not the case for *ad hoc* Wi-Fi where the beaconing system results in the unit being rather too fully awake for most of the time to save on power.

Further investigation has revealed that we are actually dealing with a rather more fundamental problem here. It appears that all the situations where long battery lifetime has been achieved are asymmetrical. They rely either on the receiver node being permanently awake and receptive to short 'discovery pings' from the other node or being synchronized with the pinging node. In the first case the node making the short pings can become more or less totally quiescent between pings. In the second case the receiver node need only wake up for a small fraction of the time to see if a ping is being transmitted. With suitable design both of the radio scheme and the device, this can make the average power consumption of the pinging device and/or the receiver device very small indeed. The scenarios envisaged for Low Energy Bluetooth as well as the usual relationship between cellular telephones and their base stations are all asymmetrical, allowing the battery lifetime of the client side to be kept very high. *Asymmetric Discovery* with long battery lifetime is now well established.

What we were trying to achieve in N4C was a more symmetrical situation where two unsynchronized peer nodes are trying to discover each other's existence but both would be as near possible quiescent when not actively transmitting. Conventional wisdom would indicate that it might be possible to

manage this by keeping the receiver circuitry permanently active but transmitting only rarely, using a low power system. Regrettably this is not currently the case. For the sophisticated radio systems used in Wi-Fi and Bluetooth, the differential between power consumed in receive and transmit modes is relatively tiny (perhaps 100mW), whereas the background consumption in both modes may be 5-10 times as high. Long battery lifetime is achieved by shutting down the radio (and as much else) as possible when not required and ensuring that the radio can transition from the off state to transmitting very rapidly so that power is not wasted in a long stabilization phase.

The object in N4C was to have the nodes operate totally independently with no close synchronisation and discover each other as they pass close by. With readily available radio technology this appears to be impossible. Thus *Symmetric Discovery* cannot be deployed in a long term test bed, although small scale experiments can be performed and N4C has shown how the system might work.

It appears that solving the Symmetric Discovery problem requires a radio technology solution which is outside the scope of N4C. Finding a solution that pairs a micropower receiver that can be kept permanently awake with an occasional burst transmitter such as is used in Low Energy Bluetooth might be the key. In CCRs and disaster recovery scenarios, Symmetric Discovery would seem to be highly desirable and we believe that this is an important topic for future study.

17. EVALUATION OF ARCHITECTURE AND FUTURE WORK

The software architecture used for DTN nodes and the overall architecture of the DTN systems as documented in Part 1 of this report has proved satisfactory for the experiments carried out during N4C. The experiments have taken place over a period of more than 2½ years and, in the case of the Slovenian test bed, are still continuing. During this time well in excess of 10,000 opportunistic encounters between node pairs have occurred and been logged. In the summer test period of 2010 when the experiments were at the highest point the village router DTN2-based system logged over 5000 encounters and the LTU Prophet-based system logged in excess of 2000 actual node encounters. A very large number of bundle transfers have taken place as can be seen from the test reports from the two test beds.

The main areas of change and improvement in the architecture during the project have been related to system management, power management and logging. As described in Section 10, security was addressed other than through the use of the BSP and the Security sub-system attached to the BPA, which was the original intention as expressed in the DoW. Our view of how to achieve AAA in a DTN has also altered during the project as explained in Section 10.7.

The problems that we encountered during the experiments were, for the most part, matters of detailed design and implementation. Most of these we have been able to solve: the spiral development model meant that solutions were found and deployed in later periods of testing. The exception was the Symmetric Discovery issue discussed in Section 16. Finding a solution to this issue was beyond the scope of N4C but it seems to be an important area of work that should be investigated in an appropriate research venue. However, one related issue that was a constraint on the efficiency of connection opportunity capacity was the length of time taken to establish Wi-Fi connections when an opportunity was detected. This is not so much of a problem in a lightly loaded network, but in some cases in the Slovenian test bed it was a constraint on the volume of data that could be transferred. It was also exacerbated by the need to re-establish connections in DTN2 after they were dropped because they were idle.

Having seen how a practical DTN behaves over an extended period of trials, there are a number of architectural level work items that we believe could be usefully investigated in the future:

- Further development of the DTN naming (dtn: URI) scheme. This is linked to the information centric networking studies that are part of the SAIL project (see Section 10.7). Investigating how an analogue of the DNS system in the CI can be provided for DTNs and other CCR mechanisms.
- Implementation of PRoPHET version 2 in DTN2 and further work to investigate how the model represented by the DPs evolves over longer periods and with more nodes in the network. The integration platform developed during N4C by partner ITTI is a key tool for these experiments.
- Finding a solution to the Symmetric Discovery problem. As discussed in Section 16, this may well be a radio technology problem rather than a networking problem. Finding a solution to this problem would seem to be highly desirable for first responders in disaster recovery situations where infrastructure has been destroyed and no temporary replacements have yet been deployed.
- Providing a standardized management system for DTN nodes, using a well known MIB and communications between clients and the management system over the bundle protocol.
- Completing the implementation of the BSP in DTN2 and experimenting with practical applications of the BSP. DTN management is likely to be an area where the BSP is needed, especially if remote configuration as well as status reporting is involved.
- Investigating whether transport mechanisms such as the Bundle Protocol intended for HERTB situations and existing protocols intended for LERTB can be offered through a common interface. Even if an application is not intended to run in FMNs, having a common interface would potentially make the writing of applications simpler and more consistent.
- Investigating how much awareness of the elasticity of RTB or otherwise should applications have? Should they be explicitly informed of transitions between CI and DTN connectivity? What criteria would be used to determine when a transition has occurred?
- LERTB can be considered as a degenerate case of HERTB. It might be possible to conceal the transitions between environments within a transport designed for the high elasticity environment. Whether this is desirable needs to be discussed because of the different characteristics of the transports.

PART 3: REFERENCE INFORMATION

18. GLOSSARY

Term	Explanation	
AAA	Authentication, Authorization and Accounting	**10.6, 10.7**
Active Registration	*Registration* to which bundles can currently be delivered.	**3.2.1.3**
API	Application Programming Interface	
Application Programming Interface	Set of routines available over an RPC interface that allows applications to use the capabilities of the *BD*.	
BAB	Bundle Authentication Block – data used in the *BSP* to authenticate a bundle across a single hop in a DTN.	
Block	*Bundles* are constructed from a sequence of a *Primary* Block and one or more other *Blocks*. *Blocks* have a *Canonical Block Header* format that identifies the block type.	
Bluetooth	Personal Area Networking communication technology.	
Bundle	Basic unit of transmission of data in *DTNs* specified in [RFC5050],	
BP	*Bundle Protocol*	
BPA	*Bundle Protocol Agent*.	**[RFC5050], 3.2.1**
BSP	*Bundle Security Protocol*	**[BSP],3.2.1.3, 3.2.6**
Bundle Fragment	See *Fragmentation*.	
Bundle Protocol	Fundamental protocol used to transfer *Bundle* data units in a *DTN* as endorsed by the *IRTF DTN Research Group*. See [RFC5050]	
Bundle Protocol Agent	The bundle protocol agent of a node is the node component that offers the BP services and executes the procedures of the bundle protocol [RFC5050].	**[RFC5050], 3.2.1**
Bundle Security Protocol	Specification of additional protocol capabilities that provide data integrity and confidentiality services for the *Bundle Protocol*.	**[BSP], 3.2.1.3, 3.2.6**
CCR	*Communication Challenged Region*	
CCSDS	*Consultative Committee on Space Data Standards*	
CF	Compact Flash – non-volatile memory card format.	

Term	Explanation	
CI	*Connected Internet*	
Communication Challenged Region	Area with little or no communications infrastructure, and probably little or no permanently installed electricity supplies.	
Connected Internet	The conventional Internet where nodes are in continuous contact with neighbours according to a relatively stable topology	
Consultative Committee on Space Data Standards	Standards development organisation for the space community.	
Convergence Layer	Component that implements the functionality required to mediate between the *BPA* and an underlying transport network using a specified transport protocol and type of communication endpoint.	**3.2.3, [TCPclayer], [UDPclayer]**
Custody	Contract offered with respect to a *Bundle* by a node implementing a *DTN BD*. If a node accepts *Custody* of a *Bundle*, it contracts to maintain the node in reliable, persistent storage for the *Bundle's* specified lifetime and seek to forward the *Bundle* either to another node that will accept *Custody* or to its final destination *EID*. In effect a node accepting *Custody* becomes a surrogate for the original source node.	
Delivery Predictability	An estimate of the probability that a *Bundle* will be delivered to its destination if it remains in this *BD* node. Used in the [PRoPHET] routing algorithm.	
Destination EID	Field in the *Bundle Primary Block specifying the EID* of a node to which the *Bundle* is to be delivered.	**3.2.1.3, [RFC5050]**
DHCP	Dynamic Host Configuration Protocol	
Diagnostic Interplanetary Network Gateway	Protocol for remote management of the *BD* over a transport using *Bundles* and a *MIB* to organize the data carried.	
DING	*Diagnostic Interplanetary Network Gateway* protocol.	
DNS	Domain Name System	
DoS	Denial of Service – security attack that works by impeding the legitimate users' access to a service, typically by overloading it.	
DoW	Description of Work – Annex 1 of the N4C Proposal document.	
DP	*Delivery Predictability*	
DTN	Delay- and Disruption-Tolerant Networking.	

Term	Explanation	
DTN2	Reference implementation of a *BD* that implements a large part of the functionality described in this document, and more besides. Sponsored by *DTNRG*.	
DTNRG	Internet Research Task Force *DTN* Research Group. Sponsors of the *DTN2* reference implementation from which this specification is derived.	
Dynamic Routing	Category of *DTN* routing and forwarding mechanisms where routing decisions are influenced by information that is shared during opportunistic encounters of nodes (such as *Delivery Predictabilities*) in networks which have little or no predetermined topology rather than being controlled by a static table of routes or exchange of topology information in a network that has a relatively static topology.	
EID	Endpoint Identifier.	
Elasticity	Term applied to how variation an application might need to cope with in the *RTB* depending on the type of network in which the node deploying the application finds itself. See *HERTB* and *LERTB*.	
Endpoint	A *Bundle Endpoint*, usually known as just an *Endpoint* in this document and elsewhere, is a set of zero or more nodes supporting a *BD* to and from which *Bundles* can be sent. These nodes all identify themselves for all purposes associated with the *Bundle Protocol* by a single text string, the *Endpoint Identifier*. Note that a node may identify itself by multiple *EIDs* and hence be a member of multiple *Bundle Endpoints*.	
Endpoint ID	Endpoint Identifier	
Endpoint Identifier	Identifier for a *Bundle Endpoint*. Takes the form of a Uniform Resource Identifier (URI) [RFC3986]. It is intended that all *EIDs* will be taken from the *dtn:* URI scheme currently under development in the *DTNRG*.	
Extension Block	Generic name for *Bundle Blocks* other than the *Primary Block* and the *Payload Block*. At present *DTN2* categorizes *Extension Blocks* into *Metadata Blocks*, *Bundle Security Protocol Blocks*, and *'others'*. There are several sorts of *other* blocks defined already, and it may be necessary to improve this categorization in future.	
FIRE	Future Internet research and Experimentation initiative. European Union experimentally driven research programme of which N4C is a part.	

Term	Explanation	
Flood Routing	*Flood* or *Epidemic Routing* involves sending a copy of every *Bundle* that arrives at the *BD* to every open *Link* and forwarding a copy of every extant, non-expired *Bundle* on every *Opportunistic Link* that is discovered and opened.	
Fragmentation	*Bundles* with a *Payload Block* that is longer than one octet, can be fragmented into two *(Bundle) Fragments* that can be *Forwarded* separately. Each *Fragment* must have at least one octet in its *Payload Block*. The rules about which *Blocks* are placed in each *Fragment* are described in [RFC5050] and later in this document. On arrival at the *Destination Endpoint*, *Bundle Reassembly* is performed on the *Fragments* to reconstitute the original bundle.	
GSM	Global System for Mobile Communications – second generation mobile cellular telephone system.	
HERTB	*High Elasticity Round Trip Bound*	
High Elasticity Round Trip Bound	A class of network such as DTN where a node or application can expect to encounter a wide variation in the *RTB* over time and depending on the route that data takes through the network.	
HTTP	HyperText Transfer Protocol	
IAB	Internet Architecture Board	
IETF	Internet Engineering Task Force	
IMAP	Internet Message Access Protocol – *CI* protocol used to communicate between email servers and *MUA*s.	
Internet Draft	Pre-standard draft document published by the Internet Engineering Task Force.	
IP	Internet Protocol.	
IPsec	IP layer security system	
IRTF	Internet Research Task Force	
Key Database	Database of security *Keys* maintained for use by the *BSP*.	
LERTB	*Low Elasticity Round Trip Bound*	
Licklider Transmission Protocol	Transport protocol designed to provide retransmission-based reliability over links characterized by extremely long message round-trip times (RTTs) and/or frequent interruptions in connectivity.	

Term	Explanation	
Link	Potential (available or unavailable) or Actual (opening or open) communication channel using a specific *Convergence Layer* to a *Next-hop* node.	
Low Elasticity Round Trip Bound	Class of network in which the *RTB* is restricted to a small range of relatively low values. Characteristic of the conventional Internet (*CI*).	
LTP	*Licklider Transmission Protocol.*	
Management Information Base	Scheme for structuring management and configuration information used for the *DING* and *SNMP* protocols.	
Maximum Transmission Unit	Constraint on the size of data unit that can be transmitted over a *Convergence Layer*.	
Metadata Block	*Bundle Protocol Extension Block* designed to carry additional information that DTN nodes can use to make processing decisions regarding bundles, such as deciding whether to store a bundle or determining to which nodes to forward a bundle.	
MIB	Management Information Base	
MUA	Mail User Agent – client application used for sending and receiving email and providing a human user interface.	
MX	Mail eXchanger record – entry in a DNS zone specifying the IP address of the machine that accepts *SMTP* connections for the delivery of email.	
N4C	Networking for Communication Challenged Communities. The EU Framework Programme 7 project that supported the creation of this document.	
NAT	Network Address Translator	
OpenVPN	Open source implementation of a *VPN* system.	
Opportunistic	Description of a class of scenarios in *DTNs* in which communication between nodes occurs only when they happen to meet in the course of their life, usually according to some *Mobility* Pattern. Also used for a type of *link* that opens a *contact* either in response to a direct request from the proposed peer or as a result of the reception of an *Announcement* from the proposed peer. Once opened the *Contact* remains open until explicitly closed or communication with the peer is broken.	
Passive Registration	*Registration* to which *bundles* cannot currently be delivered.	

Term	Explanation	
Payload	The user data or PDU carried in a *Bundle*.	
Payload Block	The *Block* that encapsulates the *Payload* in a *Bundle*. A *Bundle* has either zero or one *Payload Blocks* and the *Payload* has to have at least one octet of data.	
PCB	Payload Confidentiality Block – data used in the *BSP* to provide parameters needed to decrypt an encrypted bundle payload after arriving at a security destination node.	
PDA	*Personal Digital Assistant*	
PDU	*Protocol Data Unit*.	
Persistent Storage	Permanent storage used to hold *Bundles*, etc. to allow the *BD* to be shutdown and restarted without loss of information.	
Personal Digital Assistant	Pocket sized portable computer system with useful applications for daily life. Appropriate applications will often depend on the work or leisure occupations of the owner. (This term has had a remarkably short lifetime! Now usually a mobile telephone with installed applications… think iPhone® Apps!)	
PIB	Payload Integrity Block – data used in the *BSP* to provide confirmation of the integrity of the bundle payload after arriving at a security destination node.	
PII	Personally Identifying Information	
Primary Block	Master *Block* that must be the first *Block* in a *Bundle*.	
Prophet	Implementation of this architecture with restricted features but using the PRoPHET routing protocol (compare *DTN2*).	
PRoPHET Routing	*DTN Dynamic Routing* mechanism using *Delivery Predictabilities*.	
Protocol Data Unit	The data of the next higher layer protocol carried in a protocol message.	
Registration	Expression of interest in *Bundles* with a particular set of *Destination EIDs*. When an application is running, it may make a *Registration* into an *Active Registration* that results in *Bundles* with matching *Destination EIDs* being delivered to the application. Otherwise *Registrations* are *Passive Registrations*.	
Registration ID	Unique number identifying a *Registration*. Preserved across *BPA* shutdown and restart until the *Persistent Storage* database is reinitialized.	

Term	Explanation	
RFC	Request for Comments. IETF and IRTF publications containing standards and related documents.	
RFCOMM CL	*Bluetooth Connection Oriented Convergence Layer.*	
Round Trip Bound	The maximum time expected to be taken by a message making a return journey between two nodes (and the source and destination applications). See also *Elasticity.*	
RTB	*Round Trip Bound*	
SBC	Single Board Computer	
Security Policy Database	Repository of information indicating what *BSP* mechanisms should be applied when sending *Bundles* to a particular *Destination EID* and what security mechanisms should have been applied on incoming *Bundles.*	
Security Policy Enforcement Point	The *BD* provides enforcement of security as specified by the *SPD* and the *BSP* for *Bundles* received and transmitted by the *BD*, i.e., it provides a *Security Enforcement Point.*	
Service Tag	String concatenated with an *EID* for a node (probably part of the URI *path* component) to provide a demultiplexing mechanism for *Bundles* arriving at a node that is identified by the *EID.*	
Simple Network Management Protocol	Protocol used in the conventional Internet to carry Network Management information.	
SNC	Sámi Networking Connectivity. Predecessor project to N4C.	
SNMP	*Simple Network Management Protocol.*	**15**
Source EID	Field in the *Bundle Primary Block* specifying an *EID* of the node from which the *Bundle* has been sent. May be the *Null EID.*	**3.2.1.3, [RFC5050]**
SPD	*Security Policy Database.*	
SPEP	*Security Policy Enforcement Point*	
SSH	Secure SHell – secured tunnel remote access system.	
SSL	Secure Socket Layer	
Static Routing	Uses a routing table filled only by configuration commands. Therefore the routing provided is all statically defined.	
Status Report	*Administrative Bundle* with the data format of a *Status Report* as defined by [RFC5050] as its *Payload.*	

Term	Explanation	
Store, Carry and Forward	Fundamental paradigm of *DTN*.	
TCP	Transport Control Protocol - Connection oriented transport protocol in the IP suite.	
UDP	User Datagram Protocol. Part of the IP suite.	
Uniform Resource Identifier	Structured identifier as used for naming Endpoints. Constructed from a *Scheme* name (DTN expects to concentrate on the DTN-specific *Scheme dtn:* being defined in [URIscheme] and [URIfind]) and a *Scheme Specific Part* being the remainder of the URI.	
URI	*Uniform Resource Identifier*.	
VPN	Virtual Private Network	
Wi-Fi/802.11	Wireless local area networking technology.	
WiMAX/IEEE 802.16	Wireless networking technology similar to but with considerably greater range than *Wi-Fi*.	
WP	Work Package	
ZigBee/IEEE 802.15.4	Wireless networking technology suitable for low power personal area networks (PANs).	

19. REFERENCES

[Ahlgren10]	B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher and B. Ohlman, *"A Survey of Information-Centric Networking,"* Dagstuhl Seminar Proceedings, Dagstuhl seminar 10492, ISSN 1862-4405, November 2010.
[AltTime]	Farrell, S, McMahon, A., and Ott, J., *"Handling Issues with Real Time in the Bundle Protocol"*, draft-farrell-dtnrg-alt-time-00, November 2009.
[BPMIB]	Sims, Z., *"Bundle Protocol MIB"*, draft-sims-dtnrg-bpmib-00, *March 2011.*
[BSP]	Symington, S., Farrell, S., Weiss, H., and Lovell, P., *"Bundle Security Protocol Specification"*, draft-irtf-dtnrg-bundle-security-15, February 2010.
[CCSDS]	The Consultative Committee for Space Data Systems.
[Checksum]	Eddy, W., Wood, L., and Ivancic, W., *"Reliability-only Ciphersuites for the Bundle Protocol"*, draft-irtf-dtnrg-bundle-checksum-06, October 2009.
[Ding]	Clark, G., Campbell, G., Kruse, H. and Ostermann, S., *"DING Protocol -- A Protocol For Network Management"*, draft-irtf-dtnrg-ding-network-management-02, February 2010.
[DoW]	N4C Consortium, *"Networking for Communications Challenged Communities: Architecture, Test Beds and Innovative Alliances: Grant Agreement Annex 1: Description of Work"*, Framework Programme 7 Grant Agreement Fp7-223994, April 2008 – January 2011
[DTN2]	Demmer, M., *et al*, *"DTN2 Bundle Protocol Agent Reference Implementation"*, Available from Sourceforge DTN Repository or N4C Code Repository.
[DTNatTCD]	DTN web page at Distributed Systems Group. Trinity College Dublin.
[DTNcode]	Sourceforge code repository for DTN2 and associated code. Project page.
[DTNinSpace]	CCSDS, *"Rationale, Scenarios, and Requirements for DTN in Space"*, Informational Report CCSDS 734.0-G-1, August 2010.
[DTNRG]	Internet Research Task Force (IRTF) Delay- and Disruption-Tolerant Research Group Web Site and Wiki.
[DTNstateArt]	Davies, E, *et al, "DTN - The State of the Art"*, N4C Deliverable D2.1, April 2009,
[DTNnetMgmt]	Ivancic, W., *"Delay/Disruption Tolerant Networking - Network Management Requirements"*, draft-ivancic-dtnrg-network-management-reqs-00, June 2009.
[EndPoint]	McMahon, A., and Fall, K., *"The Delay Tolerant Networking Endpoint Discovery Protocol"*, draft-mcmahon-dtnrg-dtn-edp-00, February 2010.
[Epidemic]	Vahdat, A., and Becker, D., *"Epidemic Routing for Partially Connected Ad Hoc Networks"*, Duke University Technical Report CS-200006, April 2000.

[EPOXI]	EPOXI (Extrasolar Planet Observation and Deep Impact Extended Investigation) spacecraft (previously known as *Deep Impact*), Press Release on DTN experiments
[Fall08]	Fall, K., and Farrell, S., *"DTN: an architectural retrospective"*, IEEE J. Sel. Areas Commun., vol. 26, no. 5, pp. 828–836, 2008.
[Farrell06]	Farrell, S., and Cahill, V., *"Delay and Disruption Tolerant Networking"*, Artech House Publishers, 2006, ISBN: 1-59693-063-2.
[FuncSpec]	Davies, E, and Doria, A., *"Functional Specification for DTN Infrastructure Software"*, N4C Deliverable D2.2, April 2010
[InfCenNet]	Kutscher, D., and Farrell, S., *"Towards an Information-Centric Internet with more Things"*, Submissions for IAB SmartObjects Workshop, February 2011.
[Inquiry]	Bluetooth Special Interest Group, *"Bluetooth Communication Topology - Inquiry Procedure"*, Bluetooth SIG Web Site.
[ISSDTN]	International Space Station DTN Experiments Summary
[LTPcl]	Burleigh, S., *"Delay-Tolerant Networking LTP Convergence Layer (LTPCL) Adapter"*, draft-burleigh-dtnrg-ltpcl-03, February 2011.
[Maildir]	Maildir format specification
[Mercurial]	Mercurial source control management system
[MetadataBlock]	Symington, S., *"Delay-Tolerant Networking Metadata Extension Block"*, draft-irtf-dtnrg-bundle-metadata-block-07, February 2010.
[N4Carch]	Davies, E, *et al, "N4C System Architecture"*, N4C Deliverable D2.1, April 2009,
[N4CappEval]	N4C Consortium, *"D3.2 Evaluation and progress report"*, Deliverable 3.2, February 2010
[N4Ccode]	N4C code repository
[NodeDesign]	N4C Consortium, *"N4C Node Design"*, Deliverable 5.1, May 2009.
[One]	"The ONE" The Opportunistic Network Environment simulator.
[PRoPHET]	Lindgren, A., Doria, A., Davies, E., and Grasic, S., *"Probabilistic Routing Protocol for Intermittently Connected Networks"*, draft-irtf-dtnrg-prophet-05, February 2010.
[ProphetDTN]	Grašič, S., *et al.*, *"Implementation of PRoPHET DTN Routing Protocol"*, Available from N4C Code Repository, as made for Sámi Network Connectivity project.
[QueryBlk]	Farrell, S., Lynch, A., Kutscher, D., and Lindgren, A., *"Bundle Protocol Query Extension Block"*, draft-farrell-dtnrg-bpq-00, November 2010.

[RAPID]	Balasubramanian, A., Levine. B., and Venkataramani, A., *"DTN Routing as a Resource Allocation Problem"*, ACM SIGCOMM'07, August 27–31, 2007, Kyoto, Japan.
[RFC3588]	P. Calhoun, et al, *"Diameter Base Protocol,"* Internet RFC 3588, September 2003. RFC 3588
[RFC3971]	Arkko, J., Kempf, J., Zill, B., and Nikander, P., *"SEcure Neighbor Discovery (SEND)"*, RFC 3971, *March 2005*
[RFC3972]	Aura, T., *"Cryptographically Generated Addresses (CGA)"*, RFC 3972, March 2005.
[RFC3986]	Berners-Lee, T., Fielding, R., and Masinter, L., *"Uniform Resource Identifier (URI): Generic Syntax"*, STD 66, RFC 3986, January 2005.
[RFC4838]	Cerf, V., *et al*, *"Delay-Tolerant Networking Architecture"*, RFC 4838, April 2007.
[RFC5050]	Scott, K. and Burleigh, S., *"Bundle Protocol Specification"*, RFC 5050, November 2007.
[RFC5322]	Resnick, P., *"Internet Message Format"*, RFC 5322, October 2008.
[RFC5325]	Burleigh, S., Ramadas, M., and Farrell, S., *"Licklider Transmission Protocol - Motivation"*, RFC 5325, September 2008.
[RFC5326]	Ramadas, M., Burleigh, S., and Farrell, S., *"Licklider Transmission Protocol - Specification"*, RFC 5326, September 2008.
[RFC5327]	Farrell, S., Ramadas, M., and Burleigh, S., *"Licklider Transmission Protocol - Security Extensions"*, RFC 5327, September 2008.
[RFCOMM]	Bluetooth Special Interest Group, *"RFCOMM - How it works"*, Bluetooth SIG Web Site.
[RFCQueue]	RFC Editor Publication queue
[SAIL]	*"Scalable and Adaptive Internet"*, FP7 SAIL project, http://www.sail-project.eu/ Accessed 2011-03-11
[SDNV]	Eddy, W. and Davies, E., *"Using Self-Delimiting Numeric Values in Protocols"*, draft-irtf-dtnrg-sdnv-09, February 2011.
[SlvSummer2009]	MEIS d.o.o., *"M8.3 Summer 2 interconnected tests report, Version V05"*, Document: n4c-wp8-004-M8.3-summer_test_2009_Slovenia_V05_and_QAQC_V50.pdf, January 2010.
[SolarPwrMgt]	Hartnett, K., *et al*, *"DTN node power management and performance during the N4C summer 2010 Trial", To be published, April 2011.*
[Spiral]	The Spiral Development Model Originally developed by Professor Barry Boehm in 1986.

[SquirrelMail]	SquirrelMail web email interface
[SwdWinter2011]	Grašič, S. *"M8.7 Winter test report 2011 (Swedish test field)"*, Document: M8.7_N4C-WP8.7-1.2.pdf, March 2011.
[TCPclayer]	Demmer, M. and Ott, J., *"Delay Tolerant Networking TCP Convergence Layer Protocol"*, draft-irtf-dtnrg-tcp-clayer-02, November 2008.
[Thunderbird]	Mozilla Thunderbird Mail User Agent
[Karvo08]	Karvo, J., and Ott, J., *"Time Scales and Delay-Tolerant Routing Protocols"*, ACM CHANTS'08, September 15, 2008, San Francisco, California, USA.
[UDPclayer]	Kruse, H. and Ostermann, S., *"UDP Convergence Layers for the DTN Bundle and LTP Protocols"*, draft-irtf-dtnrg-udp-clayer-00, November 2008.
[URIfind]	Davies, E. and Doria, A., *"Adding the "find" Operation to the dtn: URI Scheme"*, draft-davies-dtnrg-uri-find-01, October 2009.
[URIscheme]	Fall, K., Burleigh, S., Doria, A., and Ott, J., *"The DTN URI Scheme"*, draft-irtf-dtnrg-dtn-uri-scheme-00, March 2009.

The following three items are provided as separate documents.

ANNEX 1: SYSTEM INTEGRATION PLATFORM OVERVIEW

ANNEX 2: DATA ANALYSIS TOOLS

ANNEX 3: PROPHET PROTOCOL WORK IN DETAIL