

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación



**PROPUESTA Y SIMULACIÓN DE UNA
SOLUCIÓN BASADA EN REDES
TOLERANTES AL RETARDO PARA
PROPORCIONAR COMUNICACIONES EN
ENTORNOS REMOTOS AISLADOS**

TRABAJO FIN DE MÁSTER

Jhon Eduardo Bedoya Camacho

2013

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación

**Máster Universitario en
Ingeniería de Redes y Servicios Telemáticos**

TRABAJO FIN DE MÁSTER

**PROPUESTA Y SIMULACIÓN DE UNA
SOLUCIÓN BASADA EN REDES
TOLERANTE AL RETARDO PARA
PROPORCIONAR COMUNICACIONES EN
ENTORNOS REMOTOS AISLADOS**

Autor

Jhon Eduardo Bedoya Camacho

Director

Manuel Álvarez-Campana Fernández-Corredor

Departamento de Ingeniería de Sistemas Telemáticos

2013

Resumen

La Internet ha sido un gran éxito en los dispositivos de comunicación y de interconexión en todo el mundo. Esto se ha logrado mediante el uso de un conjunto de protocolos de comunicación normalizados, entre los que destacan TCP e IP, junto al despliegue de una infraestructura de miles de redes y subredes que encaminan todo el tráfico de este y otros protocolos. En el diseño de redes TCP/IP los criterios originales de la red suponen alta disponibilidad, control distribuido, conmutación de paquetes, datagramas, independencia de redes y heterogeneidad. Otra suposición habitual es que el retardo de tránsito a través de la red es moderado. Sin embargo, en paralelo al crecimiento de Internet, han surgido nuevas arquitecturas de red y nuevos desafíos para llevar a servicio a todos los usuarios.

En este contexto, dentro de los desafíos está lograr llevar algún tipo de comunicación en entornos remotos, queriendo decir con remotos en donde no hay disponibilidad y una arquitectura de red donde soporte algún protocolo de red conocido para internet. De este modo, la tecnología DTN (Delay Tolerant Network), ha surgido como una solución para operar redes en entornos hostiles,

En este documento se realiza una evaluación de la tecnología DTN principalmente orientada a estudios para llevar Internet en ambientes remotos aislados utilizando el modelo de un nodo móvil el cual transporta y entrega mensajes a otros. Para validar la solución, se hace uso de un entorno de simulación de DTNs ampliamente utilizado en el mundo académico: The ONE. Mediante dicho simulador se evalúan aspectos tales como el protocolo de encaminamiento, el ratio de entrega de mensajes a sus destinos, la latencia de los mismos o el modelo de movilidad de los nodos DTNs. De este modo, se trata de evaluar el comportamiento de la solución diseñada antes de ser implementada en un entorno real.

Abstract

The Internet has been a great success in communication devices and interconnection all around the world. This has been achieved by using a set of standard communication protocols, among them TCP and IP, in addition to the deployment of an infrastructure of thousands of networks and subnetworks that route all traffic of these and other protocols. In the design of TCP/IP networks the original criteria involve high availability network, distributed control, packet switching, datagrams, independence and heterogeneous networks. Another common assumption is that the transit delay through the network is moderate. However, together with the growth of the Internet, new network architectures and new challenges have been created to bring service to all users.

In this context, among the challenges is to reach some kind of communication in remote, meaning where there's no availability and network architecture to support any known network protocol for the Internet.

Thus, DTN (Delay Tolerant Networking) technology has emerged as an adequate solution for enabling communications in challenging scenarios.

This document provides an assessment of DTN technology mainly oriented to studies for bringing the Internet to isolated and remote scenarios, proposing a communication model where a mobile node carries and delivers messages to others. To validate the solution, we make use of a simulation environment for DTNs widely employed in academia: The ONE. Using this simulator, we evaluate aspects such as the routing protocol, the latency and message delivery ratio to their destinations, or the mobility model of the DTN nodes. Thus, it comes to assessing the performance of the designed solution before implementation in a real environment.

Agradecimientos

Especiales agradecimientos a todas las personas que me han brindado su apoyo y voz de aliento durante todo el tiempo de realización del Máster, a mi familia y a mi tutor por la orientación y ayuda con material durante la realización del trabajo para cumplir el objetivo.

Índice general

.....	
1	Introducción..... 1
1.1	Motivación..... 2
1.2	Objetivos..... 5
1.3	Estructura de la Memoria..... 5
2	Estado del Arte 7
2.1	Origen de las redes DTN..... 7
2.2	Impacto del retardo en TCP/IP 8
2.3	Ámbitos de aplicación de las DTNs..... 10
2.3.1	Conectividad en las zonas en desarrollo, entornos remotos 10
2.3.2	Comunicaciones espaciales 12
2.3.3	Redes tácticas militares..... 12
2.3.4	Monitorización en entornos ambientales 14
2.3.5	Redes de sensores 16
2.3.6	Redes con dispositivos móviles..... 17
2.4	Software para DTNs..... 18
2.4.1	Librerías y entornos de desarrollo 18
2.4.2	Simulación y Emulación..... 19
3	Arquitectura y Protocolos DTNs..... 21
3.1	Tipos de nodos..... 21
3.2	Método de almacenamiento y reenvío 22
3.3	Tipos de contactos 24
3.4	La Capa Bundle 25
3.5	Protocolos de Encaminamiento..... 32
4	Entorno de Simulación The One 35
4.1	Estructura del simulador..... 35
4.2	Modelos de movilidad..... 36

4.3	Ejecución de la simulación	38
4.4	Creación de los Mapas	40
4.4.1	Definición de escenarios de simulación	41
5	Simulación del caso de estudio	43
5.1	Especificación del escenario a simular	43
5.1.1	Configuración de los nodos	44
5.1.2	Configuración de movilidad de los nodos.....	44
5.1.3	Generación de Mensajes	45
5.2	Selección de Reports	46
6	Resultados de la simulación	48
6.1	Reportes	49
7	Conclusiones	53
8	Trabajo Futuro	53
	Bibliografía	55

Índice de figuras

Figura 1. Red DakNet.....	11
Figura 2. Red Kiosknet.....	11
Figura 3. Condor Gateway.....	13
Figura 4. Red DTN con modelo CONDOR.....	14
Figura 5. Arquitectura EMMA.....	15
Figura 6. Red Bytewalla.....	17
Figura 7. Dispositivo Android con Software IBR-DTN.....	19
Figura 8. Nodos DTNs.....	22
Figura 9. Almacenamiento y Reenvío en las DTNs.....	23
Figura 10. Capas Protocolo Bundle.....	26
Figura 11. Modelo de Custodian Blundes.....	30
Figura 12 Estructura Simulador The One.....	36
Figura 13 Modelo de Movimiento del Paquete.....	38
Figura 14 Interfaz de Visualización del Simulador.....	39
Figura 15 Ejemplo de Creación de Líneas	40
Figura 16 Ejemplo de un ejercicio Interfaz final.....	41
Figura 17 Mapa del Escenario.....	43
Figura 18 Modelo Epidemic Procotol.....	45
Figura 19 Interfaz Grafica de Propuesta de Red en The One.....	48
Figura 20 Tiempo de Creación Mensajes hacia las Aldeas.....	49

Figura 21 Reporte DistanceDelayReport.....	50
Figura 22 Reporte MessageDelayReport.....	51
Figura 23 Reporte MessageStatusReport.....	52

Siglas

BP	Bundle Protocol
DARPA	Defense Advanced Research Projects
DTN	Delay Tolerant Network
DTNRG	Delay-Tolerant Networking Research Group
EPLRS	Enhanced Position Location Reporting System
IETF	Internet Engineering Task Force
GCBA	Commercial Generic Bioprocessing Apparatus
LAN	Local Area Network
MANET	Mobile Ad Hoc Networks
ONE	Opportunistic Network Environment Simulator
UWB	Ultra-Wide Band
WAN	Wide Area Network
WSN	Wireless Sensor Networks

1 Introducción

Los escenarios de red más habituales normalmente suponen conexiones estables, ideales en donde las arquitecturas de red funcionen óptimamente para la transmisión y el enrutamiento de datos. Cuando esto deja de suceder y surgen problemas como lo son pérdidas, retardos, latencias entre otros, entonces las arquitecturas con las que se están soportando ya no son suficientes para llevar a cabo la tarea de la comunicación.

El retardo y las interrupciones en la red son efectos que llevan a que se introduzca un nuevo concepto de arquitectura de red llamado *Delay- Disruption Tolerant Network* (DTN). El objetivo de este es llevar la transmisión de los datos en ambientes donde estos efectos se presentan. Los retrasos actuales de extremo a extremo en la Internet tienden a medirse en milisegundos y hasta microsegundos según el tipo de red, ya sea redes de área local (LAN), redes de áreas extensas (WAN), inalámbricas (Wireless) y satelitales, donde llegan a los segundos.

Por último, la arquitectura de redes DTN comienza a tener presencia cuando ya no es posible que el mensaje llegue a su destinatario o cuando los retardos son muy altos, un ejemplo donde se presentan estas fallas son en los siguientes escenarios de los cuales se explicara más adelante:

- Redes interplanetarias
- Conexiones entre satélites
- Villas o lugares remotos de difícil acceso
- Fauna animal
- Comunicaciones vehiculares
- Redes de sensores

DTN cuenta con el apoyo de Vint Cerf, que ha usado con éxito el sistema para transmitir datos a través del espacio, uno de los padres de la actual Internet, en EE.UU., DARPA y NASA están financiando un número de proyectos de investigación sobre DTNS, recientemente la Agencia Espacial Europea (ESA) y la NASA, en manos del comandante Sunita Williams de la Estación Espacial Internacional controló un robot construido por LG y situado en la Tierra desde dicha estación vía DTN durante 90 minutos, e intercambió datos con la máquina a una velocidad de bajada de 50 Kbps y de 82 bytes de subida. La transmisión de los datos de un punto a otro llevó siete segundos. La Agencia espera poder utilizar DTN en las misiones espaciales que se lancen a partir de 2015. Pero además, como la nueva arquitectura está diseñada para tratar conectividad intermitente, podría utilizarse en la Tierra con diferentes propósitos, como en redes militares con dispositivos móviles que entran y salen continuamente de la cobertura de la red. [1]

En Europa se han realizado otros proyectos, uno de esto es N4C el cual trabaja en llevar internet en ambientes de difícil acceso [2], cuenta con el apoyo de Dr. Stephen Farrell unos de los principales investigadores en DTNs, además de eso, el Grupo de Investigación de DTN IRTF (Grupo de Investigación de Internet Fuerza) se ha producido un importante número de RFCs y borradores de Internet, estos se incluyen la especificación de un Arquitectura DTN RFC [3], basada en el intercambio asíncrono de mensajes arbitrariamente del tamaño de paquetes entre extremos de la comunicación.

La arquitectura DTN se define como una interconexión de red de superposición diferente a internet con pilas de protocolos subyacentes arbitrarias. El protocolo capa Bundle [4], que opera por encima de la de transporte capa, es el responsable de reenviar los paquetes desde el origen al destino en un salto de hub a hub, tras un enfoque de conmutación de mensajes (store-and-forward).

1.1 Motivación

Existen muchas regiones que no tienen acceso a cualquier tipo de conexión convencional para que puedan conectarse con el mundo de la Internet. Es este momento la tecnología DTN otorgan la posibilidad de conexiones en estos

ambientes remotos o de difícil acceso, además es un campo de investigación de plena actualidad ya se ha expuesto en varias conferencias y estudios realizados y publicados por varias organizaciones dedicadas a ello. En opinión las redes tolerantes a retardos pueden significar una gran solución a varios desafíos presentes en entornos remotos, además la simulación es una gran herramienta para el análisis en una ambiente determinado.

Delay-Disruption Tolerant Network ó Redes Tolerantes a Retrasos es el nombre asignado por el grupo de investigación del IETF, aquellas redes donde se presenta retardos extremos, difícil acceso o instalar una arquitectura que funcione con algunos de los protocolos conocidos de internet, es un nuevo paradigma de comunicación que pueden extenderse a través de múltiples redes y hacer frente a las duras condiciones no previstas en el Modelo de Internet, con el fin lograr llevar algún servicio o aplicación a un punto final. Las redes DTN son una arquitectura basada en middleware de Internet independiente, donde protocolos en todas las capas se utilizan más apropiadamente para su operación dentro de cada entorno, con un nuevo protocolo de red (Bundle Protocol) que se inserta entre las aplicaciones y las comunicaciones optimizadas a nivel stack de protocolos.

Las redes DTNs soportan interoperabilidad con otras redes acomodando largas demoras entre y dentro de las regiones mientras que transcurren en este ambiente. Las DTNs han surgido recientemente como una nueva tecnología para ofrecer una solución eficaz a las comunicaciones de escenarios en condiciones extremas como las que suele manifestarse en MANETs (alta movilidad, alto retrasos, comunicaciones esporádicas, etc.).

Actualmente las redes DTN se han implementado en varias áreas tales como comunicaciones de espacio profundo con retardos de propagación extremadamente largos, comunicaciones vehiculares con contactos oportunistas entre los nodos móviles, redes de vehículos rurales en entornos remotos, redes de sensores para control de fauna animal, redes tácticas, comunicaciones espaciales entre otros. En estas situaciones, los servicios de comunicaciones pueden enfrentar la pérdida impredecible de cobertura y retrasos en la transferencia de los datos.

Básicamente las DTNs operan en donde los escenarios no cumplen los habituales supuestos IP:

- Tasas de error elevadas por el uso de enlaces inalámbricos, por lo general no puede soportar el tipo de conectividad de extremo a extremo requerido por el modelo clásico TCP e IP (redes ad-hoc, redes de sensores, redes satelitales,..)
- Comunicaciones intermitentes:
 - Contactos esporádicos en nodos móviles en casos planificados
 - Apagado y encendido de nodos ayuda al ahorro de energía
 - Desvanecimiento de señal en enlaces de radio

Ya se ha mencionado, en algunos ambientes no funcionan los protocolos de internet, ya que existen supuestos de red como:

- La pérdida de extremo a extremo es relativamente baja.
- Todos los routers y estaciones terminales soportan los protocolos TCP e IP.
- Las aplicaciones no tienen que preocuparse por el rendimiento de comunicación.
- La conmutación de paquetes es el método más adecuado para la interoperabilidad y el rendimiento.
- Selección de una sola ruta entre el emisor y el receptor es suficiente para lograr aceptables rendimiento de la comunicación.

La arquitectura DTN supone realizar mejoras a los anteriores supuestos, basándose en un diseño con los siguientes principios:

- Utiliza una sintaxis de nombres que es compatible con una amplia gama de convenios de denominación y direccionamiento para mejorar interoperabilidad.
- El mecanismo utilizado para proveer estas conexiones es el de almacenamiento y renvió, donde un nodo almacenar un mensaje el tiempo necesario hasta que exista una conexión disponible para enviarlo a un nodo siguiente, no requieren fiabilidad de extremo a extremo.
- El almacenamiento es suficientemente persistente y robusto para almacenar hasta que el renvío de paquetes puede ocurrir, e (implícitamente) que este modelo "store-and-forward" es una mejor elección de tratar de efectuar una conectividad continua.

1.2 Objetivos

Los objetivos del presente trabajo son fundamentalmente:

- El estudio de la tecnología DTN para ofrecer servicios de conexión en entornos remotos, enfocando principalmente en el modelo de red que cuenta con un nodo móvil para llevar servicios a estos lugares.
- Utilización de la herramienta The One para realizar propuesta de simulación expuesta en el punto anterior.

1.3 Estructura de la Memoria

Este trabajo se encuentra dividido en 8 capítulos, el primero de ellos, en el cual nos encontramos, ofrece una visión del trabajo y una visión general de las redes DTNs, que se ha realizado. En el segundo capítulo se realiza una revisión de estado del arte de las redes DTN, aplicaciones realizadas y software utilizado para estas implementaciones, el tercero entra de manera más profunda y se aborda el tema que son las redes DTN, como funcionan, protocolos y enrutamiento utilizados para ello, en el cuarto capítulo se expone la estructura del simulador que se va a utilizar y como se genera un entorno de simulación, el quinto capítulo ya se profundiza en si la implementación a realizar se describe la configuración del escenario de red, configuración de nodos y demás parámetros utilizados, además se encuentra la parte de los resultados obtenidos en gráfica luego de haber ejecutado dicha simulación, el sexto capítulo se documentan los resultados obtenidos y el análisis de ellos, en los últimos capítulos realizamos las conclusiones de los resultados obtenidos y la importancia de ellos, trabajos futuros para mejorar los resultados obtenidos y agradecimientos.

2 Estado del Arte

2.1 Origen de las redes DTN

La mayoría de protocolos de transporte, como TCP se basan en la suposición de que el remitente y el receptor están conectados continuamente por alguna ruta de transmisión, si el protocolo falla y los datos no pueden ser entregados. En algunas redes, no existe una ruta de extremo a extremo. Un ejemplo es una red espacial como LEO (órbita baja de la Tierra) los satélites pasan dentro y fuera del alcance de las estaciones terrestres. Un satélite dado puede ser capaz de comunicarse con una estación de tierra sólo en determinados momentos, y dos satélites nunca pueden ser capaces de comunicarse entre sí en cualquier momento, incluso a través de una estación ubicada en tierra, la razón es que alguno de estos puede encontrarse fuera de rango es decir está apuntando hacia otro lugar. Otro tipo de redes son conexiones para: submarinos, autobuses, teléfonos móviles y otros dispositivos con que los equipos no conectan de forma constante debido a la movilidad o las condiciones extremas. En este tipo de redes se conectan ocasionalmente, los datos aún pueden ser comunicados mediante el almacenamiento en los nodos y renviarlos más tarde, cuando no hay un enlace disponible. Esta técnica es llamada conmutación de mensajes. Luego finalmente, los datos se transmiten al destino. Una red cuya arquitectura se basa en este enfoque se denomina DTN (Network Delay-Tolerante, o redes tolerantes al retardo)

El trabajo en DTNs comenzó en 2002, cuando Internet Engenieer Take Force o El Grupo de Trabajo de Ingeniería de Internet IETF creó un grupo de investigación sobre el tema, la inspiración para DTNs provenía de la idea: los esfuerzos para enviar paquetes en el espacio.

Desde 2002, la arquitectura de DTN ha mejorado, y las aplicaciones del modelo DTN han crecido, Por consiguiente, el Grupo de Investigación de Internet (IRTF) ha creado un nuevo grupo de investigación para examinar la manera en general de DTN - Este grupo es llamado el DTNRG[17], y está actualmente enfocado a trabajar en la arquitectura y protocolos DTN.

El DTNRG desarrolla dos protocolos principales, el Bundle Protocol (BP) [4], y Licklider Transmission Protocol (LTP), [18]. El grupo DTNRG documenta estos protocolos en las RFCs, y deben someterse a la implementación de Internet. DTN es una tecnología nueva y hay áreas bastante grandes, donde queda mucho por hacer antes de poder implementar esta tecnología, en cualquier lugar de la escala de la arquitectura e incluso una parte de la Internet actual.

2.2 Impacto del retardo en TCP/IP

El protocolo TCP/IP provee el transporte de conexiones de la mayoría de aplicaciones existentes, estas conexiones realizan transporte de información entre diferentes dispositivos ejemplo un PC y servidor web localizado en cierto punto, para lograr esto se hace desde una red local LAN, o WLAN, etc. Mediante un ISP y finalmente llegue a un servidor web.

El modelo de servicio de Internet basado en TCP / IP existente proporciona conexión de extremo a extremo utilizando la unión de las tecnologías de la capa de enlace. La estandarización del protocolo IP y su asignación en marcos específicos de la red de datos de la capa de enlace de cada router soporta interoperabilidad mediante un modelo de conmutación de paquetes de servicio. Así se garantiza el rendimiento general del servicio entre las rutas existentes de extremo a extremo, el tiempo máximo de ida y vuelta entre cualquier nodo en la red no es excesivo y es mínima la probabilidad de que se descarte un paquete. Pero esto no es suficiente en todos los escenarios donde se necesite tener acceso a internet, surge un nuevo desafío de redes, que no cumplen con uno o más de estos supuestos, las cuales son muy importantes y pueden no estar bien atendidos por el actual protocolo de extremo a extremo con el modelo TCP / IP.

Las transmisión en TCP tiene el modelo de "SYN-ACK" desde un servidor a un punto final, mientras estas conexiones se mantengan en los milisegundos existirá mínimas fallas en la transmisión de mensajes, ahora cuando comienzan a superar el orden de los 2 minutos ya no puede haber establecimiento, a medida que el retardo aumenta, los flujos TCP normales comienzan a recibir

con menor eficacia, debido al hecho de que la espera de una ida y vuelta se hace más y más larga e ineficaz ya que la cantidad de retardo de ancho de banda se hace más grande. El estándar TCP es un protocolo que realmente se limita al uso en casos de latencia más bajos tal vez hasta un satélite en órbita geoestacionario donde los retrasos son del orden de un cuarto de un segundo.

Para explicar por qué TCP presenta estos problemas para ciertos escenarios de aplicación, podría ayudar a proporcionar un ejemplo. Gran parte del trabajo tiene sus raíces en un proyecto de investigación de la NASA para desarrollar una Red Internet interplanetaria (IPN). La idea básica es tratar de hacer que hubiese comunicación de datos entre la Tierra y un punto remoto una nave espacial. Hay un tiempo de espera de genéricos para las conexiones, de dos minutos implementado en la mayoría de pilas TCP: si no se envían datos o reciben en este tiempo, la conexión se interrumpe. Poner estos hechos juntos, podemos ver que, una nave espacial que esta a más de un minuto de distancia, todo intento de establecer una conexión TCP conexión fallará, y no se va a transmitir datos nunca de la aplicación. En el caso de Marte, por ejemplo, en su máxima aproximación a la Tierra, el round-trip times (RTT) es aproximadamente ocho minutos con el peor de los RTT de aproximadamente 40 minutos. Por lo tanto, TCP no puede funcionar en absoluto para las comunicaciones Tierra-Marte. Por supuesto, la comunicación con naves espaciales nunca será tan fácil como utilizar el medio terrestre de Internet, ya que muchas otras dificultades se deben superar.

2.3 Ámbitos de aplicación de las DTNs

Como ya se ha mencionado, los protocolos que se usan tan ampliamente a través de Internet pueden no ser aplicables a todos los tipos de redes, existen varios escenarios donde operan bajo las limitaciones de los retardos y pérdidas estos casos son:

2.3.1 Conectividad en las zonas en desarrollo, entornos remotos

La conectividad de red y de Internet para aplicaciones o servicios humanos de comunicaciones en zonas aisladas incluye varios desafíos, sobre todo referentes a la disponibilidad de infraestructura y costos asociados con la instalación y el mantenimiento de los equipos que se requieren para desplegar el servicio. Se han desarrollado varios proyectos con este modelo a continuación algunos de ellos:

- **DakNet:** Es un desarrollo con el modelo DTN para proporcionar acceso de red a los pueblos rurales de la India y Camboya [5]. Fue el primero en introducir el concepto de "datos por mula", es decir, un medio en línea, puede ser un autobús, una motocicleta, o incluso una bicicleta, la idea es mover datos entre puntos desconectados. Utiliza una red ad hoc basándose en tecnología inalámbrica para proporcionar conectividad digital asíncrona, este proyecto ha logrado implementar la arquitectura a un costo menor que las soluciones de red fija. (Figura 1).
- **KioskNet:** Es una implementación piloto en ella se instala un conjunto de kioscos los cuales KioskNet los cuales utilizan mecánico backhaul Seth. [6], como el principal medio de comunicación a Internet (Fig. 2). Los Ferries transportan los datos hacia y desde un kiosco para un conjunto de gateways que se comunican con un proxy en Internet.

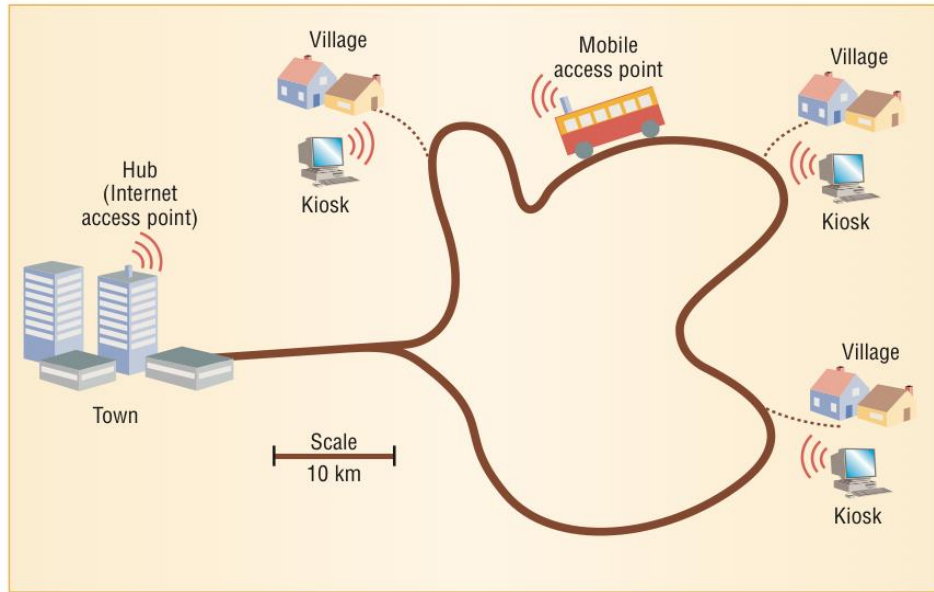


Figura 1 Concepto DakNet. Transporte físico, un autobús público, lleva un punto de acceso móvil (MAP) entre los kioscos de aldea y un hub con acceso a Internet fuente (5).

— N4C [7]: Es otro proyecto con el objetivo de que exista comunicación a internet en sitios remotos, la que no es posible instalar alguna infraestructura conocida por el costo que representa, pretende crear una arquitectura de red oportunista en estas áreas remotas para aprovechar el despliegue de gente los cuales podrían llevar los datos a ciertos puntos. Uno de los sitios de pruebas se encuentra en Laponia Municipio sueco de Jokkmokk y Gällivare, el otro lugar de la prueba se encuentra en las montañas de Eslovenia en la región de Kočevje.

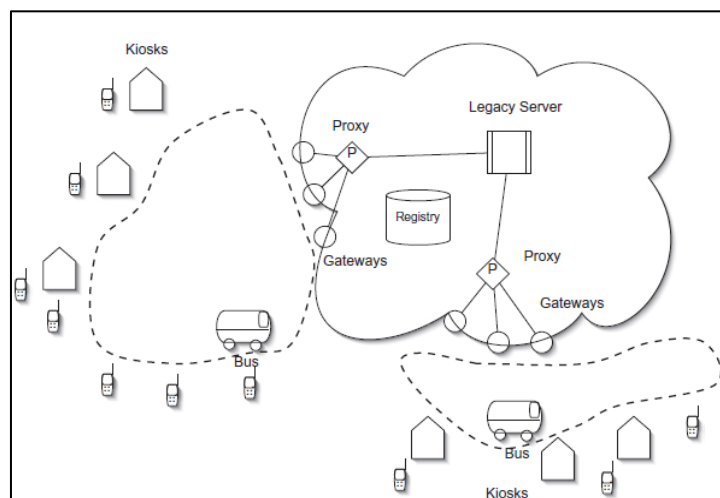


Figura 2 Red Kiosknet fuente (6).

2.3.2 Comunicaciones espaciales

Son medios de comunicación no comunes, incluye comunicaciones cercanas a la Tierra por satélite, enlaces de radio de larga distancia o (por ejemplo, comunicaciones ópticas de espacio profundo con la propagación de la luz, retrasos en segundos o minutos), básicamente los desarrollan los organismos del gobierno dedicados a pruebas de algún tipo de conexión en el espacio o hacia este, de ahí se ha derivado el concepto de DTN que se abordara en otro capítulo, actualmente la NASA trabaja sobre el de Programa Networking Tolerante la interrupción (DTN), el cual establece un compromiso a largo plazo, con el fin de tener facilidad de acceso las comunicaciones del banco de pruebas que esta a bordo de la Estación Espacial Internacional (ISS). Se instalan dos dispositivos, Comercial Generic Bioprocesamiento (GCBA), GCBA-5 y GCBA-4, los cuales servirán como equipos de prueba de comunicación que transmiten mensajes entre ISS y los Centros de Control de Misión en tierra. Todos los datos serán monitoreados y controlados en el Centro de Operaciones de Carga Útil BioServe remoto Control (POCC) ubicado en las instalaciones del Centro de Ingeniería de la Universidad de Colorado - Boulder, [8] la arquitectura DTN DTN permite mejorar la transmisión de datos durante traspasos de la estación, maximizar la utilización de eficiencia del enlace y minimizar costos.

2.3.3 Redes tácticas militares

Estos sistemas pueden operar en ambientes hostiles donde la movilidad, los factores del medio ambiente, e interferencia intencional pueden ser causa de la desconexión. Para las DTN estos entornos son ideales ya que la Blunde Layer actuar como unificador en la parte superior de las capas de convergencia de enlaces y tecnologías de redes.

El cuerpo de la Marina con el proyecto de CONDOR desarrolla una infraestructura fuerte para los desafíos de maniobras que perturban la comunicación, ya que los efectos de interrupción son persistentes, estas interrupciones causan inestabilidad en la sincronización de las bases de datos.

La arquitectura DTN ayuda a CONDOR en varios aspectos entre ellos esta lograr mantener una comunicación en algunos periodos de conectividad, además de ser fiable, obtención imágenes en sitios de acceso difícil, correo electrónico entre otros.

Básicamente la red DTN CONDOR consiste, en un CONDOR Gateway (vehículo) tiene incluido un router cisco 3725, y un radio EPLRS, un módulo de cifrado de enlace, un router de acceso Cisco, y un terminal de comunicación por satélite. El software DTN2 es portado a CONDOR y se cargan en el módulo IDS de Cisco (un ordenador de una placa con un procesador Intel Pentium III, 512 MB RAM, 20 GB de disco duro, y una interfaz Ethernet), [9]. (Figura 3), (Figura 5).

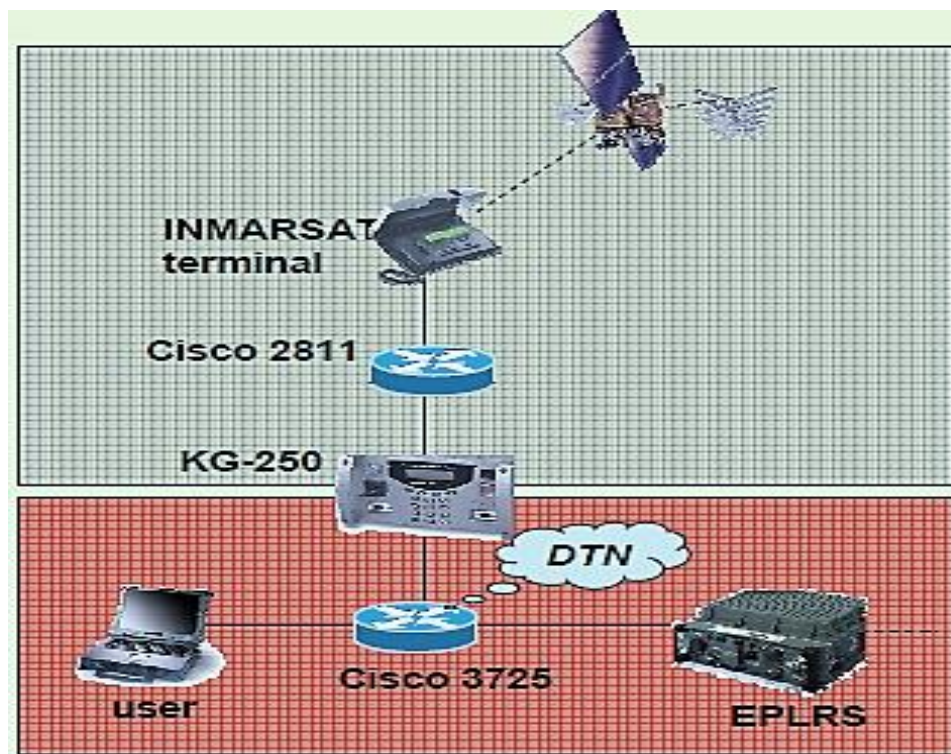


Figura 3 Estructura de Condor Gateway fuente (9).

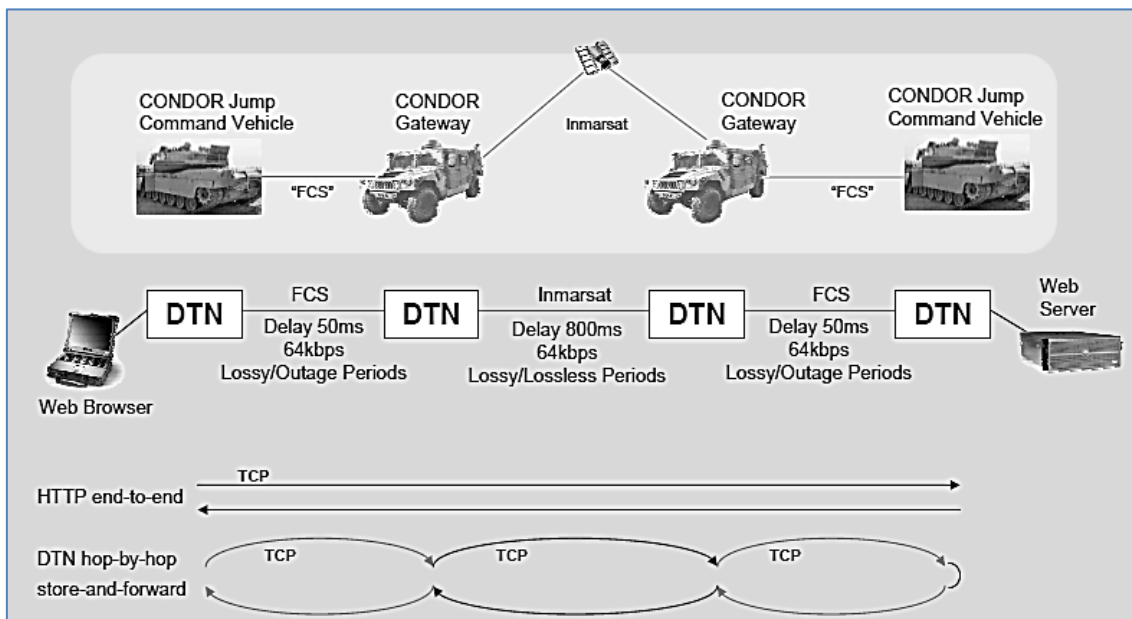


Figura 4 Red DTN con modelo CONDOR. Fuente (9)

2.3.4 Monitorización en entornos ambientales

- Zebranet:** Las redes en la fauna animal se instalan para la supervisión de población en un área determinada, el seguimiento de los datos en los sensores instalados en los animales van a ser recogidos diariamente o semanalmente por una estación base instalada en un vehículo, avión (llamado un "ferry"). Estos datos de seguimiento también se replican cuando hay contacto entre otros animales. La idea nace básicamente en el año 2002 y pretende recoger información sobre el comportamiento social y el movimiento de las cebras en un área de 100 kilómetros cuadrados en la población de Kenia [10].
- Monitoreo Ambiental en Áreas Metropolitanas-EMMA:** Es un proyecto con el fin de vigilar el medio ambiente en áreas metropolitanas, toma mediciones contaminantes en el aire en sectores urbanos, el proyecto utiliza los vehículos de transporte público, tales como autobuses y tranvías, para recoger continuamente datos del medio ambiente, la información que se intercambia entre los vehículos por medio de WLAN y estos utilizan Técnicas de DTN, a fin de minimizar los costos de transmisión. Posteriormente los valores

tomados se envían a una central Gateway que hará comunicación con el servidor principal y allí se evaluará los datos recogidos. [11] Figura 5.

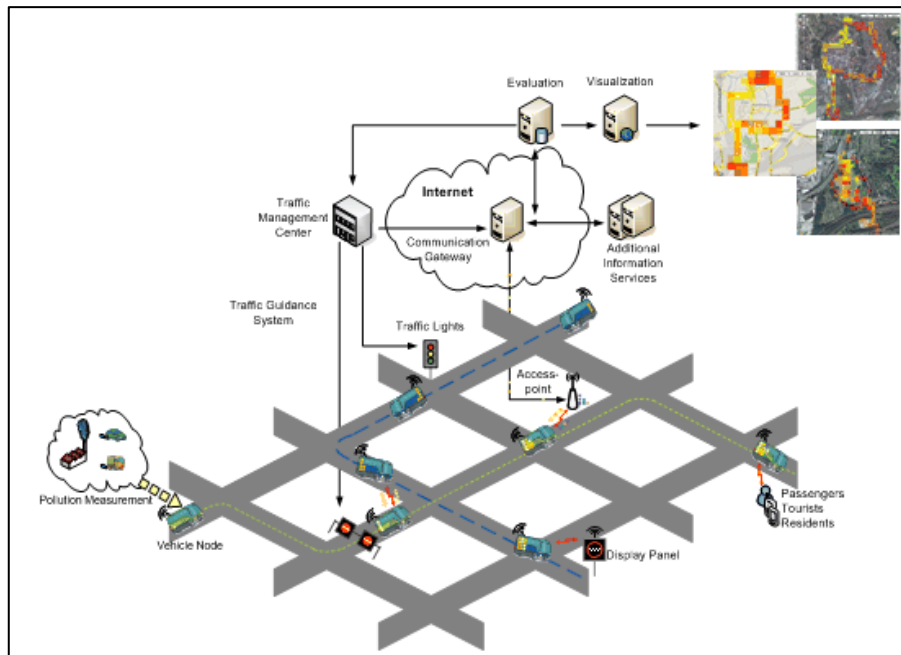


Figura 5 Arquitectura EMMA Fuente (11)

- Monitoreo en entornos remotos con adquisición de datos en una red basada en sensores autónomos (MANA):** El proyecto MANA abordó el problema de la implementación de una red de sensores para la monitorización del lago durante todo el año en el noreste Groenlandia [12]. El Monitoreo terrestre en el Alto Ártico es difícil debido a las condiciones extremas, hace que el acceso sea difícil y limitado, además de la ausencia de infraestructura de comunicación. Un sistema de adquisición de datos de redes de sensores basados en Zackenberg, Groenlandia, aborda estos desafíos.

Existen muchos más proyectos en este área podemos nombrar algunos otros por ejemplo: monitoreo sísmico, control de fuego en ambientes forestales, redes submarinas acústicas,¹ estos ambientes son en la mayoría de difícil acceso y arquitectura difícil y costosa de instalar, las DTN ayudan a que estos costos bajen suficientemente en vez de utilizar una tecnología con de teléfonos móviles para transmisión de los datos.

¹ S. Farrell and V. Cahill, Delay- and Disruption-Tolerant Networking. Norwood, MA, USA: Artech House, Inc., 2006

2.3.5 Redes de sensores

Las redes de sensores por lo general están limitadas en cuanto a la alimentación de energía, Memoria, capacidad de almacenamiento, comúnmente los programas son diseñados teniendo en cuenta estos factores, por lo general se utilizan nodos llamados "proxy" para traducir los protocolos de internet y así funcione con la arquitectura diseñada.

Los dispositivos mencionados anteriormente forman redes, dependiendo de las características, necesidades del diseño, tecnología de comunicaciones, protocolos de encaminamiento de mensajes, etc. En los últimos años se ha introducido un nuevo concepto de red llamado WSN (Wireless Sensor Network). Según [13]:

Una red de sensores inalámbricos (WSN) consiste en un gran número de pequeños nodos, con capacidad de cómputo limitada, que cooperan en tareas de detección y se comunican con tecnologías de comunicaciones inalámbricas.

Las redes de sensores inalámbricos (WSN) son considerados como buena aplicación de DTN; una WSN puede ser una "región" o un área en una infraestructura DTN la cual usa gateways DTN para transferir datos a través de la Internet.

Los servicios que pueden ofrecer las WSN son bastantes, podemos destacar algunos de los más importantes:

- Aplicaciones de monitorización de hábitats naturales.
- Observación de entorno y sistemas de previsión.
- Aplicaciones relacionadas con la salud.
- Aplicaciones militares.
- Monitorización de edificios inteligentes.
- Sistemas inteligentes para control del tráfico.
- Sistemas domóticos y entornos inteligentes.
- Redes Vehiculares

2.3.6 Redes con dispositivos móviles

Los dispositivos móviles también son utilizados en la arquitectura DTN, tanto para redes sociales, sistemas de recomendación o en ambientes remotos para llevar internet a ellos. Pueden emplear múltiples facilidades y oportunidades de conexión pero no es fácil su desarrollo, para lograr ello depende el software y las especificaciones del equipo, además pueden aprovechar múltiples oportunidades de comunicación, como por ejemplo a través de WiFi, Bluetooth, USB, y las redes celulares 3G. En general estos dispositivos tienen un S.O especializado y / o de propiedad e implementaciones para DTN, se requiere un esfuerzo de desarrollo para proporcionar funcionalidades en DTN.

Existen varios proyectos en los cuales se ha desarrollado software e implementaciones para DTN algunos de ellos son:

- **Bytewalla:** Desarrollado por KTH- Royal Institute of Technology en Suecia, con el fin de realizar interconexión en aldeas de África y actuar como mulas de datos, con terminales Android, interface Wifi y tecnología DTN, básicamente desarrolla una implantación de la capa Blunde en lenguaje Java. Pr un DTN Bundle implementación del Protocolo en Java por KTH, Suecia para los dispositivos Android [14].

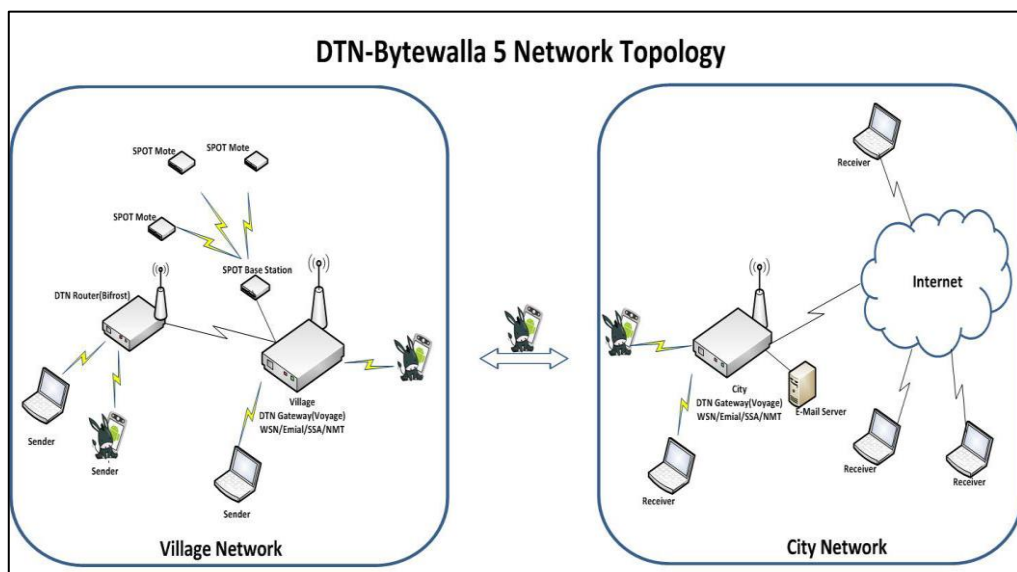


Figura 6 Red Bytewalla Fuente (14).

- **DASM:** Es una aplicación DTN para teléfonos Symbian [15], primero fue probado con el Nokia 9300i, posteriormente su desarrollo fue remplazado por DTNS60. DTNS60 es una aplicación DTN que se dirige a móviles Symbian S60, DTNS60 incluye "DT-Talkie", una aplicación DTN². DT-Talkie fue probado en Nokia N800 y N810 basado en Linux para Internet y una Tablet con S.O 2008 (Maemo 4.0).

2.4 Software para DTNs

2.4.1 Librerías y entornos de desarrollo

- **DTN2:** El objetivo de esta aplicación es reunir los componentes de la arquitectura DTN, mientras que también proporciona un framework de software robusto y flexible para la experimentación, la extensión, y el despliegue en el mundo real, [16]. Esta escrito en código C++³, y disponible para S.O Linux, Mac OSX, Win32.
- **ION:** ION (Red Interplanetaria Overlay) es una implementación del protocolo Bundle, LTP y el protocolo de entrega del archivo CCSDS (CFDP) y el servicio de mensajes asíncronos (AMS). La intención de ION es apoyar, el despliegue de pequeñas dimensiones, de alta velocidad de DTN en sistemas embebidos, tales como naves espaciales robóticas, pero también funciona en ambientes más ricos en recursos, como las redes de área local, [17].
- **IBR-DTN:** IBR-DTN es una aplicación C++ del Protocolo Bundle desarrollado en el IBR. IBR-DTN soporta la capa de convergencia TCP y UDP, [16].

² <http://www.netlab.tkk.fi/tutkimus/dtn/dttalkie/>

³ <http://sourceforge.net/projects/dtn/>

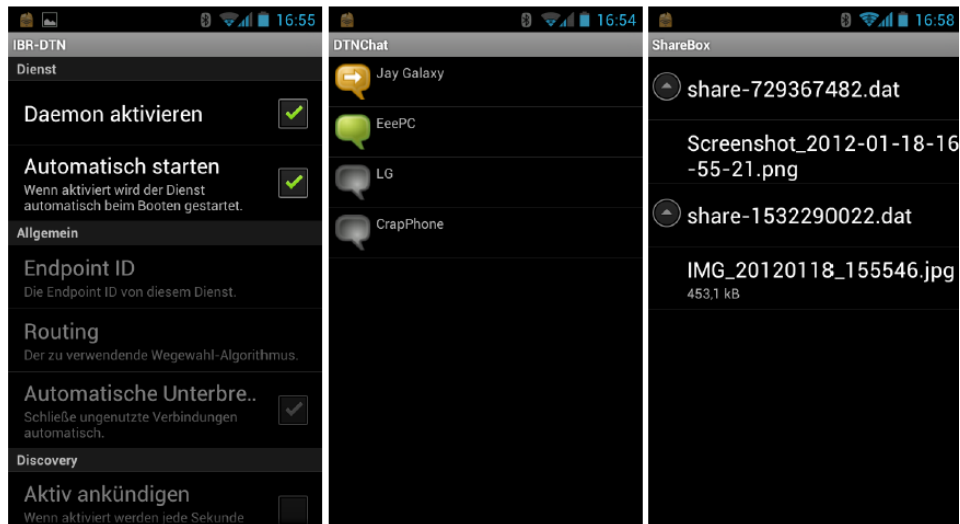


Figura 7 vista de un dispositivo Android con IBR-DTN fuente (16)

2.4.2 Entornos de simulación y emulación

La simulación y la emulación de redes DTN es necesaria ya que la mayoría de los entornos donde se desea instalar alguna infraestructura son bastantes hostiles y de acceso limitado, entonces es necesario el desarrollo de nuevas ideas y protocolos en un software antes de cualquier operación. Hay una gran variación de las capacidades y el enfoque de cada entorno, teniendo en cuenta el tamaño de las redes, por ello se debe seleccionar cuidadosamente la herramienta adecuada para adaptarse sus necesidades.

Dentro de los más importantes simuladores de enrutamiento encontramos los siguientes:

- DTNSim2 [30], este simulador esta escrito en lenguaje Java, desarrollado por la universidad de Waterloo, es un simulador exclusivo para enrutamiento DTN.
- NS-2 [31], Ns-2 es un simulador de eventos discretos orientado a la investigación en red, proporciona un apoyo sustancial para la simulación de TCP, enrutamiento y protocolos de multidifusión a través de redes de cableadas o inalámbricas (locales y por satélite).
- OMNET++ [32], es un simulador de eventos discretos basado en C++ para el modelado redes de comunicación, multiprocesadores y otros sistemas distribuidos. OMNeT++ es de código abierto, y puede ser

es usado bajo la Licencia Pública General de GNU o bajo su propia licencia.

La motivación de desarrollar OMNeT++ era producir una poderosa herramienta de simulación de eventos discretos de código abierto que pudiera ser utilizada por educadores o temas de investigación.

- The ONE [33], Opportunistic Networking Environment (ONE), es un simulador diseñado específicamente para evaluar enrutamiento en redes DTN y los protocolos de aplicación. Permite a los usuarios crear escenarios basados en diferentes modelos de movimiento o importar escenarios bajo otras aplicaciones como por ejemplo mapas de alguna área específica en donde se quiera realizar alguna simulación, ofrece un framework para la aplicación de enrutamiento y los protocolos de aplicación (ya incluyendo seis protocolos conocidos como Epidemic, ProPHET, Spray-and-Wait etc.). Posee una interfaz gráfica de visualización e interactiva y herramientas de post-procesamiento apoyar la evaluación de experimentos y un modo de emulación. Esta herramienta es la que vamos a utilizar para realizar la simulación de la propuesta de escenario propuesto del cual en el siguiente capítulo se aborda con más profundidad.

3 Arquitectura y Protocolos DTNs

Este capítulo presenta características principales para la arquitectura de las DTNs, definida por la RFC 4838. Dicha arquitectura busca hacer frente al problema de comunicación en entornos difíciles a través de del almacenamiento y reenvío de mensajes que aprovechan un conjunto de capas de red convergentes que se adaptan a una amplia variedad de transportes subyacentes. Así, las tecnologías inalámbricas para DTN pueden llegar a ser diversas, incluyendo no sólo de radio frecuencia (RF), sino también ultra-ancha banda (UWB), medios ópticos y las tecnologías de acústica (sonar o ultrasonidos).

En la arquitectura normalizada se especifican aspectos tales como los tipos de nodos y enlaces (contactos) que pueden existir en una DTN, el principio básico de operación basado en encaminamiento y reenvío, implementado dentro de la capa Bundle (normalizada en una RFC específica, la 5050), el direccionamiento basado en regiones, etc.

En el capítulo se abordan también otros aspectos no normalizados, pero imprescindibles para el funcionamiento de las DTNs, como son los protocolos de encaminamiento, o la seguridad.

3.1 Tipos de nodos

En una DTN, pueden existir tres tipos de nodos: host, router o Gateways (o alguna combinación de estos) que actúa como origen, destino o como ente reenviador de paquetes:

- Host: Envía y/o recibe paquetes, pero no los reenvía. Un host puede ser origen o destino de la transferencia de un paquete. La capa bundle en un host opera soportando largos retrasos y para esto requiere tener una capacidad de almacenamiento permanente en el cual encolar los paquetes hasta que encuentra un link disponible.
- Router: Reenvían paquetes dentro de una única región DTN y pueden opcionalmente funcionar como host. La capa bundle en los routers al igual que en los hosts operan soportando largos retrasos y para esto requieren

tener una capacidad de almacenamiento permanente en el cual encolar los paquetes hasta que encuentran un link disponible.

- Gateway: Reenvían paquetes entre dos o más regiones DTN y pueden opcionalmente ser un host. La capa bundle de los gateways deben tener almacenamiento persistente y soportar la custodia en la transferencia de los mensajes. Los gateways permiten la comunicación entre las capas inferiores del protocolo y las regiones en las cuales ellos actúan.

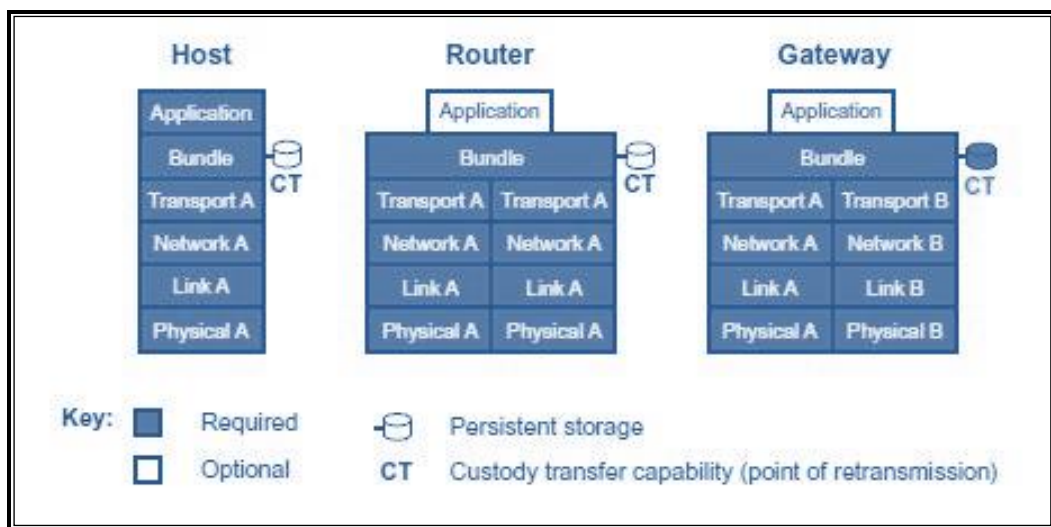


Figura 9. Nodos DTNs

3.2 Método de almacenamiento y reenvío

Cuando tenemos un enlace entre dos nodos que se comunican y están al mismo tiempo en movimiento, podemos encontrarnos con el inconveniente de que la conexión entre ambos pueda verse obstruida, ocasionando que la comunicación se inhabilite con el fin de ahorrar energía hasta que el enlace vuelva a estar disponible; esto ocasiona que la conectividad sea intermitente.

En Internet, la conectividad intermitente provoca la pérdida de datos. Los paquetes que no pueden ser inmediatamente transmitidos son normalmente desechados y TCP debe volver a retransmitirlos.

En las DTNs, por el contrario, se soporta la comunicación entre nodos con conexiones intermitentes, mediante una técnica clave para estas redes como lo es la del almacenamiento y reenvío de datos, con la cual se hace frente al problema de la intermitencia en la conectividad, los retardos variables y tasas de datos asimétricas mediante el uso del método de conmutación de datos a través del almacenamiento y reenvío, tales como los utilizados por el correo electrónico o el correo de voz.

Al contrario del modelo de Internet (en el cual los routers utilizan chips de memoria para almacenar los paquetes entrantes durante unos pocos milisegundos mientras esperan por su búsqueda del siguiente salto en la tabla de enrutamiento y un puerto disponible del router de salida), en las DTNs para que el método de almacenamiento de datos sea posible es necesario que los lugares de acopio de la información (como un disco duro) puedan almacenar los mensajes de forma indefinida. Estos son llamados “almacenadores persistentes”.

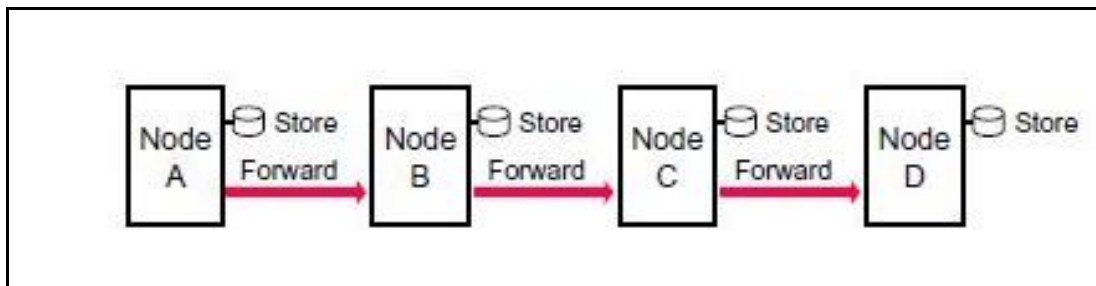


Figura 9. Almacenamiento y reenvío de datos empleado por los nodos en las DTNs

Los routers que se usan en las DTNs necesitan un almacenamiento persistente para encolar sus datos, por las siguientes razones:

- La vía de comunicación con el siguiente salto pueden no estar disponible durante un largo periodo de tiempo.

- Un nodo puede enviar o recibir datos mucho más rápidamente o de forma más fiable que el otro nodo.
- Una vez transmitidos los datos puede necesitar ser retransmitidos si se produce un error en el enlace de transmisión o si un nodo declina hacer el reenvío del mensaje.

Al mover mensajes enteros (o fragmentos de los mismos) en una única transferencia, la técnica de conmutación de mensajes proporciona a los nodos de la red un conocimiento inmediato del tamaño de los mensajes que se transfieren, y por lo tanto los requisitos para el almacenamiento de la información y el ancho de banda necesario. Dado que la constante de este tipo de redes es la movilidad debido a que los nodos suelen estar en movimiento con alimentación limitada, es fundamental que en su arquitectura se contemple el almacenamiento de datos ya que de alguna manera los datos deben resguardarse.

3.3 Tipos de contactos

Los tipos de contactos en las DTN, definidos por la RFC 4838 [3] se separan en varias categorías, basadas en sus características de rendimiento. Hasta la fecha, los siguientes tipos principales de contactos se definen de la siguiente manera:

- **Contactos persistentes.** Este tipo de contactos son los cuales están siempre disponibles para realizar la transmisión de datos, conexiones de este tipo sería una conexión de ADSL o cable modem.
- **Contactos bajo demanda.** En los contactos bajo demanda se requiere que se inicie una petición de contacto para comenzar a funcionar para luego pasar a comportarse como contactos persistentes. Ejemplo de este tipo de contactos sería una conexión de acceso telefónico ya que primero se inicia una petición de contacto que durara hasta que se termine la comunicación.

- **Contactos oportunistas.** En una red con tolerancia a retardos, los nodos involucrados, pueden necesitar comunicarse durante contactos oportunistas, en el que un emisor y un receptor hacen contacto durante un tiempo no programado, ya que lo usual es la no existencia de una ruta completa entre dos nodos que buscan comunicarse. Posiblemente los nodos de origen y de destino no estén conectados a la misma red, al mismo tiempo. Sin embargo, la habilidad que tienen las DTNs por medio de contactos oportunistas permite que dichos nodos puedan intercambiar datos entre ellos, a cambio de un retraso adicional en la entrega de los mensajes, ya que estos a menudo se almacenan en la red a la espera de una ruta disponible hacia el destino.

Vehículos, aeronaves, satélites e incluso personas (como el caso de MobiClique), pueden entrar en contacto e intercambiar información cuando, por casualidad, están dentro de la línea de visión y lo suficientemente cerca para comunicarse usando un dispositivo de comunicación.

- **Contactos programados.** En una DTN cuando tenemos un origen y destino que aun no estando en contacto directo siguen un patrón de movilidad (por ejemplo un satélite alrededor de la tierra) podemos en cierta forma planificar el envío y/o recepción de datos.

3.4 La Capa Bundle

El Bundle es la unidad básica de datos para el intercambio de información para en red DTN. Un nodo DTN puede ser un Host, Router o Gateway que actúa como origen, destino, en una red DTN, utilizando un conjunto común de capas de red (tales como TCP e IP). La Blunde Layer esta situada por debajo de la capa de aplicación como se observa en la figura 9, la información en un blunde, es muy similar a un mensaje de correo electrónico, este es luego transmitido a lo largo de una ruta que consta de una serie de máquinas intermedias que pueden cada una almacenar el paquete durante períodos significativos. Por lo tanto, la capa bundle es una red de almacenamiento y envío de protocolo. Incluye una serie de funciones de diagnóstico y gestión.

Para la interoperabilidad, utiliza un esquema de nombres flexible (sobre la base de Identificadores Uniformes de Recursos) capaces de encapsular diferentes esquemas de nombres y números en la misma sintaxis de nomenclatura general.

El protocolo bundle se describe en una serie de documentos RFCs. El primero ya antes mencionado el cual es la RFC 4838 la cual se expone la arquitectura de la red DTN en términos de la estructura y aplicabilidad, [3]. El otro documento es la RFC 5050, [4] la cual especifica el formato de los bundles y las reglas de procesamiento asociados con el envío y la recepción de los mismos. La característica mas importante de la capa bundle, es el soporte para el almacenamiento en tránsito, es decir los paquetes recibidos de un remitente puede ser almacenados en un nodo intermedio en una cantidad de tiempo excesiva (minutos, horas o incluso días). Estas operaciones son almacenadas por el stack de protocolos de la red, en la capa bundle, de forma transparente para la aplicación. El almacenamiento en tránsito es el medios para superar los retrasos y trastornos inducidos, mientras que un bundle realiza saltos hasta llegar a su destino final, para evitar costosas retransmisiones de extremo a extremo debido a los errores y tiempos de espera; y para permitir el intercambio de información entre dos nodos que no comparten ninguna trayectoria de comunicación de extremo a extremo en un momento dado momento en el tiempo.

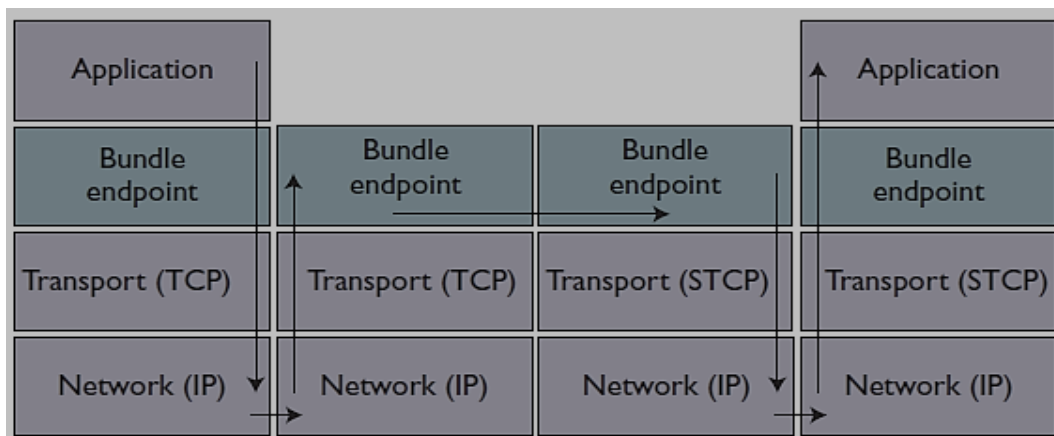


Figura 10 Protocolo Bundle. Fuente *When TCP Breaks Delay- and Disruption-Tolerant Networking*

Los problemas asociados con conectividad intermitente, largos delays, velocidad de datos asimétricos, alta tasa de error en los mensajes, son solucionados en cierta medida por las redes DTN con el modelo de Store-and-Forward o almacenamiento y envío. Cuanto los mensajes completos (bloques

enteros de datos de usuario de la aplicación de los programas) o fragmentos de este tipo de mensajes son enviados, se almacenan en algún nodo (conmutador de intersección) y luego se envían a otro lugar de almacenamiento (otro nodo) a lo largo de un camino que finalmente el mensaje llega a su destino un ejemplo de lo dicho anteriormente esta en la Figura 10.

Los Bundles tienen un formato definido que contiene dos o más bloques de datos. Cada bloque puede contener tanto datos de aplicación como otra información utilizada para entregar el bundle que contiene su destino (s). El origen y destino de los bundles se identifican por una variable de longitud, llamada Endpoint Identifiers (EIDs,) estos son los identificadores que utilizan una sintaxis como la URIs, [20] e identifican un punto final en las DTN. Un elemento esencial del estilo del bundle basado en el envío de bundles es que tienen un lugar para esperar en cola hasta que haya una oportunidad de comunicación ("contacto") disponible, esto supone que, [4]:

- El almacenamiento está disponible y bien distribuido por toda la red.
- El almacenamiento es suficientemente persistente y robusto para almacenar hasta que el envío de paquetes pueden ocurrir, y (implícitamente) que este modelo "store-and-forward" es una mejor elección de tratar de efectuar una conectividad continua.

Además los lugares de almacenamiento (como el disco duro) puede almacenar mensajes de forma indefinida, a esto se les llama almacenamiento persistente, a diferencia del almacenamiento a muy corto plazo proporcionada por chips de memoria, como que utilizan los routers de Internet, las rutas DTN necesitan un almacenamiento persistente para sus colas ya que:

- El enlace de comunicación con el siguiente salto puede o no estar disponible durante mucho tiempo.
- Puede que en un par de nodos exista una sincronización para el envío.
- Un mensaje una vez enviado puede necesitar ser retransmitido si llega ocurrir algún error.

Como se dijo anteriormente el bundle protocol está diseñado para actuar encima de diferentes tipos de redes, existen otras características en su operación en la red:

- **Regiones y Gateways:** Una región es un tipo específico de red con protocolos específicos en función de los requisitos de comunicación. Esta puede ser una red de sensores, una red militar, una red inalámbrica en un área, etc. Cada región tiene un único ID de región que es conocido con las demás regiones y forma parte de cada nombre de los nodos.

Un Gateway es básicamente, es un punto de acceso entre dos regiones que operan posiblemente en diferentes arquitecturas y pilas de protocolos, además son los únicos que pueden transportar mensajes entre varias regiones. Una vista lógica de un Gateway se observa en la Figura 10, estos son responsables de almacenar mensajes en una parte del almacenamiento persistente cuando se requiere una entrega fiable, y el mapeo entre los diferentes protocolos de transporte.
- **Nombres y direcciones:** Los puntos finales DTN son identificados por los identificadores de punto final (EID). Cada punto final ID se expresa sintácticamente como un identificador uniforme de recursos (URI). La sintaxis URI ha sido diseñado como una forma de expresar los nombres o direcciones para una amplia gama de propósitos, y por lo tanto es útil para construir nombres de puntos finales DTN. Los esquemas URI son un concepto clave en la arquitectura DTN, y ha evolucionado a partir de un concepto anterior llamado regiones, que esta mas ligado a supuestos de la topología de la red. Usando URI, se logra una gran flexibilidad en la estructuración de EID.
- **Contacto:** Los nodos no pueden ser contactados en todo momento, es necesario considerar que la comunicación es solo posible en ciertos tiempos. Cuando dos nodos no pueden comunicarse, se considera su capacidad de limite ser cero. Es decir un contacto es entonces la duración cuando la capacidad del límite de un nodo es positiva, y se puede definir el volumen del contacto como el número de bits que pueden ser transferidos durante el contacto.
- **Clases de Prioridad:** Puesto que todos los recursos en DTNs son limitados, es necesario imponer una asignación de recursos basado en prioridades y el mecanismo de entrega, la arquitectura DTN ofrece relativamente medidas de prioridad (baja, medio, alto) para la entrega de ADU (Unidades de Datos de Aplicación). Estas prioridades diferencian el tráfico basándose en el deseo de una aplicación pueda afectar a la entrega de una ADU, y se realizan en bloques de paquetes generados a causa de agrupar capas basadas en la información especificada por la aplicación.

En el documentos RFC 4838, se definen tres clases de prioridades, estas clases típicamente implican alguna secuencia de programación relativa a la priorización de los paquetes en cola de un remitente, [3].

Las aplicaciones especifican su clase de prioridad solicitada y la vida útil de sus datos para cada ADU que envían. Esta información, junto con la política aplicada en los nodos DTN que selecciona cómo se envían los mensajes y los algoritmos de enrutamiento que se utilizan, afecta en general la probabilidad y la puntualidad de la entrega de cada ADU.

La clase de prioridad de un paquete sólo se requiere para relacionarse con otros paquetes de la misma fuente. Esto significa que un paquete de alta prioridad de una fuente no podrá ser entregado más rápido (con otro de diferente calidad de servicio) que un paquete de prioridad media de una fuente diferente. Lo esto quiere decir, que un paquete de alta prioridad de una fuente se gestionará con más preferencia, a un paquete de menor prioridad enviado de la misma fuente.

- **Transferencia bajo custodia:** DTN debe soportar la retransmisión de nodo a nodo, de los datos perdidos o corruptos tanto en la capa de transporte como en la capa bundle, además del protocolo de transporte, el protocolo de “blunde layer” es el único que puede dar fiabilidad entres dos extremos DTN. Las transferencias bajo custodia están soportadas entre las capas bundle de los continuos nodos cuando la aplicación desea iniciar la conexión. Cuando un nodo desea enviar, la capa bundle de este nodo comprueba si el nodo siguiente en la ruta soporta transferencias bajo custodia. De ser así, envía el mensaje, luego el nodo emisor elimina la copia del mensaje. Es proceso es continuo el nodo que ha recibido el mensaje y es el encargado de enviarlo al próximo nodo, así hasta llegar al destino. Los bundles custodian deben almacenar hasta que el blunde custodian siguiente nodo acepte, o el TTLs de espera de bundle expire, este tiempo de tiempo de vida es mayor al tiempo de espera para reconocer el bundle. Aun así este último también es lo suficiente largo para tener una transmisión confiable. En la figura 11 observamos la arquitectura.
- **Capas de convergencia:** Las características proporcionadas por las capas subyacentes de una red DTN, como entrega fiable, conexiones (con indicaciones de fallo de conexión), control de flujo, control de congestión, puede variar considerablemente. Dado que el protocolo de renvío de paquetes asume la entrega fiable de la las capas subyacentes, puede ser necesario para aumentar la pila con una capa de convergencia específica que asegurar estas características.

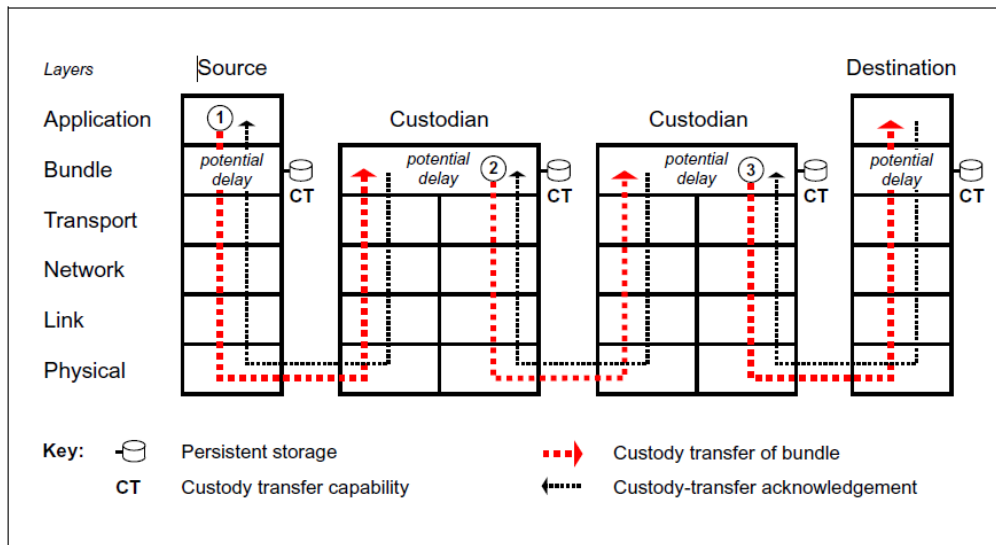


Figura 11, Modelos Custodian Blundes fuente Delay- Tolerant Network, (21)

- **Tiempo de sincronización:** La arquitectura DTN requiere la sincronización temporal entre los nodos de DTN, el protocolo Bundle asume que todos los nodos comparten un reloj común, de modo que las marcas de tiempo pueden ser interpretados y manipuladas correctamente [22], de hecho, la información de tiempo de sincronización también se utiliza para manejar caducidad la caducidad, que es análoga TTL en TCP, sino que se basa en el tiempo de creación y vida de bundle específico.
- **Control de flujo y congestión:** la congestión es un problema de gestión en lugar de una cuestión de protocolo, el acceso a los enlaces es programado y controlado, deben haber medidas para evitar y controlar la congestión, estas en general están integradas en la infraestructura de comunicaciones existentes en regiones.

Según [3], se define como "control de flujo" como el medio que asegure que la tasa promedio a la que un nodo de envío transmite los datos a un nodo receptor, no supere la tasa media a la que el nodo receptor esté preparado para recibir datos de ese remitente. Ahora "control de la congestión" es el medio de asegurar que la suma velocidad a la que todas las fuentes de tráfico inyectan datos en una red, no sea superior a la tasa global máxima a la que la red puede ofrecer los datos a los nodos de destino a través del tiempo. Si el control de flujo se propaga hacia atrás desde los nodos congestionados hacia las fuentes de tráfico, entonces el mecanismo de flujo de control

se puede utilizar como al menos como una solución parcial al problema de la congestión en sí.

Las decisiones de control de flujo DTN deben hacerse dentro de la capa bundle, esta se basa en la información acerca de los recursos (en este caso, principalmente almacenamiento persistente) disponibles en el nodo bundle. Cuando el almacenamiento recursos se vuelven escasos, un nodo DTN tiene un cierto grado de la libertad para el manejo de la situación. Siempre se puede realizar el proceso de descarte de paquetes que han expirado, al igual este proceso debe ser realizado periódicamente por los nodos DTN. Un nodo debe evitar en lo posible el descarte de paquetes para los que ha aceptado la custodia, y sólo lo hacen como último recurso.

- **Seguridad:** Está enfocada principalmente a restringir aplicaciones no autorizadas en los recursos de la red, y para prevenir autorizaciones a las solicitudes de acceso a un rango o clase superior a las que ya tienen ingreso. No es aceptable para un usuario no autorizado inundar la red con tráfico fácilmente, posiblemente negar servicio a usuarios ya autorizados, esto en muchos casos, sino que también no es aceptable reenviar tráfico no autorizado a ciertos puntos de la red. Varias consideraciones se han establecido para la seguridad DTN:
 - Evitar que las aplicaciones no autorizadas tengan sus datos almacenados o que sean llevados a través en la red DTN.
 - Evitar que las aplicaciones no autorizadas de tengan el control sobre la Infraestructura de DTN.
 - Evitar que las aplicaciones no autorizadas envíen blundles a una tipo o clase de servicio para los que carecen de permiso.
 - Inmediatamente descartar paquetes que están dañados o modificados inapropiadamente en tránsito.
- **Fragmentación:** El Protocolo Bundle permite la fragmentación de bundles, existen diferentes tecnologías y regiones que pueden manejar diferentes tamaños y mecanismos, esto hace que no puedan estar disponibles en las capas subyacentes de la red. La arquitectura DTN y el protocolo blunde no definen un mecanismo para anunciar y/ o negociar el tamaño máximo de un bundle que un nodo puede aceptar, tanto para la transmisión como para el almacenamiento [22].

Existe la posibilidad de haber ciertos instantes en que los agentes del protocolo bundle necesiten reducir el tamaño de los bundles con el fin de transmitirlos. Este podría ser el caso, por ejemplo, si el punto final para el cual un bundle el cual debe enviarse es accesible sólo a través de contactos intermitentes y el próximo contacto no tiene el tiempo suficiente para permitir la transmisión del bundle completo.

La interacción profunda del bundle fragmentado y el demás funcionalidades DTN no están claramente definidas. Se han realizado simulaciones de ello, [23]. Esto se debe en cuanto en la seguridad de los bundles esta activa en escenarios de transferencia con custodia.

3.5 Protocolos de Encaminamiento

El problema de enrutamiento en redes DTN es quizás uno de los de mayores temas de investigación, la razón es, que no hay un método exacto o preferido para manejar los problemas de enrutamiento, [24]. En primer lugar las redes convencionales de internet suponen que la topología de red esta siempre conectada (o particionada para intervalos muy cortos), el objetivo de los algoritmos de enrutamiento es encontrar la mejor ruta disponible para mover el tráfico de extremo a extremo. En una red DTN este principio no se aplica ya que las rutas de extremo a extremo no están siempre disponibles en todo momento, el enrutamiento que se realiza es para alcanzar lograr la entrega del mensaje mediante el empleo a largo plazo del almacenamiento en los nodos intermedios.

Para las redes DTN un objetivo importante es aumentar la probabilidad de la entrega del paquete, además de la reducción del retardo en la entrega también suele ser importante para las aplicaciones. La gestión del almacenamiento también está relacionado con el enrutamiento, las redes DTN poseen esquemas de encaminamiento que tienen que manejar problemas de que los nodos no están conectados constantemente, y el concepto de contacto ya a anteriormente expuesto.

La Clasificación de los algoritmos de enrutamiento DTN es algo complejo. En, [24] definen una estructura algoritmos de encaminamiento dependiendo de la cantidad de conocimiento que utilizan para calcular las rutas: algoritmos que no utilizan ninguna información de conocimiento son llamados “de

conocimiento cero”; los que utilizan información parcial para tomar decisiones llamados “de conocimiento parcial”, y los “de conocimiento completo” aquellos que tienen toda la información para el enrutamiento de los blunders. El conocimiento es el modelo a través de entidades que proporcionan, por separado, información acerca de los contactos, estado de la cola, y la demanda del tráfico, a continuación se enumeran los esquemas más utilizados, basados en la inundación de algoritmos:

1. ProPHET [25], protocolo de enrutamiento probabilístico usando la historia del encuentro y transitividad, utiliza la aleatoriedad de contactos, para replicar paquetes sólo si hay una posibilidad de para entregarlos. El protocolo de enrutamiento ProPHET se dirige principalmente a situaciones en las que al menos algunos de los nodos son móviles, básicamente cuando el nodo portador del mensaje tiene contacto con otro, se realiza la elección del siguiente nodo en la ruta de un mensaje. Al momento del contacto, el nodo portador puede decidir mediante la probabilidad de que el nuevo nodo elegido sea capaz de llevar el mensaje hasta su destino o un nuevo nodo contactado sea capaz.
2. Epidemic Protocol [26], es la distribución de mensajes de aplicación a los hosts, dentro de porciones de redes ad hoc. De esta manera, los mensajes se distribuyen rápidamente a través de porciones de la red conectada. Este Protocolo se basa en nodos que entran a la red conectada y tienen contacto, en ese momento el mensaje se propaga hacia los otros nodos, así que los mensajes tienen una alta probabilidad de llegar a su destino.
Los objetivos de epidemic son: a) maximizar la tasa de entrega de mensajes, b) minimizar la latencia de mensajes, y c) reducir al mínimo los recursos totales consumidos en la entrega de mensajes.
3. Spray-and-wait [27], este protocolo funciona de manera similar a epidemic, lo que realiza básicamente es una inundación controlada del número de copias en la red, y luego espera hasta que uno de los nodos contactados llegue a destino o contacte a alguien para hacerlo. Su

rendimiento en general es muy cercano a un esquema óptimo. Además, es altamente escalable y de buen rendimiento en una gran variedad de escenarios.

4. MaxProp [28], es un protocolo para enrutamiento de mensajes en redes DTN, este se basa en priorizar tanto la programación de paquetes transmitidos a otros nodos como la programación de los paquetes que se descarten, las prioridades se basan en probabilidades según el camino histórico de los datos y de mecanismos como reconocimientos, de los nuevos paquetes, y listas de intermediarios anterior.

5. TrafRoute [29], este protocolo tiene un esquema de direccionamiento eficiente para redes vehiculares, adecuado tanto para comunicación entre dos vehículos o desde un vehículo a una estructura, almacena una lista de los datos anteriores transcurridos para evitar la propagación de datos dos veces para el mismo nodo, además define un concepto de dominio que hace referencia a aquellos nodos que se encuentran dentro del mismo rango al punto de acceso al núcleo de la red, con este concepto clasifica el tipo de tráfico unicast en dos clases, inter-dominio e intra-dominio.

4 Entorno de Simulación The One

Se ha elegido The One como la herramienta utilizada para realizar la simulación, del cual ya se habló un poco en la sección 3.3.4, dentro de sus características vemos que ofrece un marco de simulación extensible, soporte de movilidad y generación de eventos, el intercambio de mensajes, enrutamiento de DTN y protocolos de aplicación, una noción básica del consumo de energía, la visualización y el análisis de las interfaces para la importación y exportación de los rastros de movilidad, eventos y mensajes completos.

4.1 Estructura del simulador

El entorno de simulación One se trata de un programa basado en Java que permite el realizar diferentes simulaciones de redes del tipo de las DTN, con distintos escenarios caracterizados por la movilidad que permite que se generen eventos, intercambio de datos, cambio del tamaño de los mensajes, cambio del tamaño del buffer, análisis de desempeño de protocolos de enrutamiento, permitiendo exportar los datos para luego ser evaluados.

El núcleo The One se basa como un simulador de eventos discretos basados en agentes, lo que es adecuado y suficientemente eficiente para el movimiento simultáneo y la simulación del enrutamiento.

El encaminamiento se implementa mediante ciertos módulos que deciden qué mensajes que transmiten a través de los contactos existentes. Por último, los mensajes se generan a través de generadores de eventos. La información para el enrutamiento puede ser exportada mediante otros simuladores por ejemplo dtnsim, o desde los módulos internos que manejan los algoritmos de enrutamiento, además posee dos modos para ejecutar las simulaciones, uno es la interfaz de usuario GUI y el otro es modo de comandos por lotes llamada batch mode utilizando el comando -b en el modo DOS de nuestro PC, en la figura 12 podemos observar la estructura del simulador.

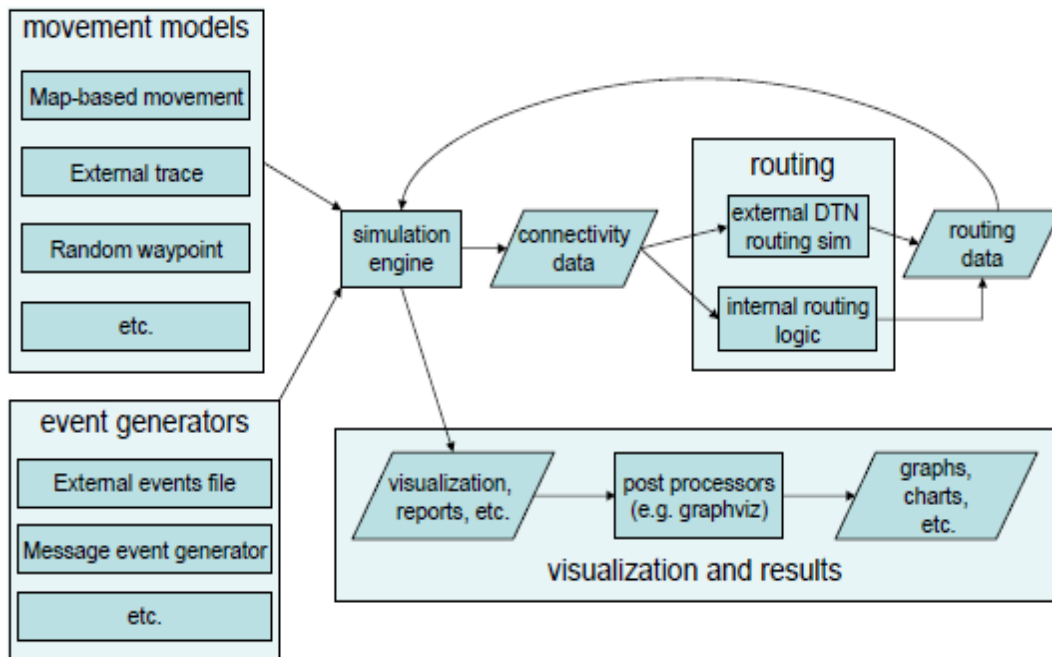


Figura 12 Estructura simulador The One Fuente (12).

4.2 Modelos de movilidad

El movimiento de los nodos se lleva a cabo bajo modelos de movimiento que vienen prestablecidos en el programa. Principalmente The One tiene cuatro modelos de movilidad, estos dictan a los nodos como deben moverse, los cuales son:

- **RandomWaypoint:** Cuando un nodo se utiliza el modelo de movimiento RandomWaypoint, dada una coordenada aleatoria en el área de simulación, los nodos se desplazan al determinado destino a velocidad constante, luego dependiendo si es necesario realizar alguna conexión se detiene por un momento, y luego crean un nuevo destino. Esto continúa a lo largo de la simulación mientras que los nodos se mueve en zig zag.
- **Map-based movement:** los modelos de movimiento basados en mapas limitan el movimiento del nodo en rutas predefinidas. Los diferentes tipos de caminos se pueden definir y The One puede definir rutas de acceso válidas para todos los grupos de nodos. De esta manera, por ejemplo, los coches pueden evitar conducir en interiores o en los

caminos peatonales. Inicialmente este modelo distribuye los nodos entre dos nodos adyacentes en el mapa (es decir, conectados por un camino) luego, los nodos comienzan a moverse desde una posición adyacente al nodo en otro mapa. Cuando el nodo alcanza el siguiente nodo del mapa, selecciona al azar el siguiente nodo adyacente del mapa pero elige el nodo del mapa dónde viene sólo si esa es la única opción (es decir, evita que se remonta a su procedencia), entonces se detiene por un tiempo y luego empieza a moverse de nuevo.

- **ShortestPathMapBasedMovement:** esta es la versión mas sofisticada del modelo basado en mapas, utiliza el algoritmo del camino más corto (Dijkstra) para encontrar su camino a través del mapa. Una vez que un nodo llega a su destino, y ha esperado el tiempo de pausa, elige un nuevo nodo de mapa aleatorio y se mueve del nodo, tomando la ruta más corta que se pueda encontrar usando sólo los nodos validos del mapa.

Este modelo de movimiento es el utiliza para nuestro nodo llamado ferry, el cual va a ser el encargado de la entrega de los mensajes a los nodos estáticos en el mapa, mas adelante se explica mejor su función.

- **MapRouteMovement:** el modelo de movimiento basado en ruta, se puede utilizar para que los nodos sigan ciertas rutas, por ejemplo, autobuses o tranvías. Por ultimo en la siguiente figura 13 se observa las clases más importantes de los modelos de movimiento.

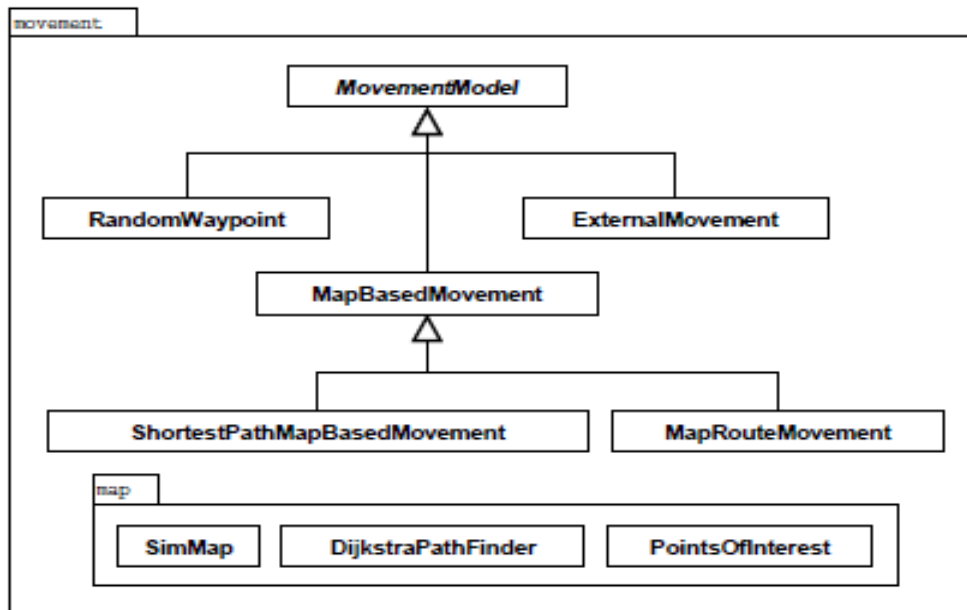


Figura 13 Modelo de movimiento del paquete fuente (35)

4.3 Ejecución de la simulación

Como se dijo anteriormente The One tiene dos modos para ejecutar las simulaciones, el modo, GUI y batch (lotes), el GUI es la interfaz gráfica de usuario modo es especialmente útil para los propósitos de prueba, depuración y demostración y el modo por lotes puede ser utilizado para el funcionamiento de gran cantidad de simulaciones con diferentes conjuntos de parámetros. Ambos modos pueden incluir cualquier número de reportes que producen las estadísticas de la simulación. Estas estadísticas se pueden analizar luego con herramientas y obtener resúmenes, gráficos y diagramas.

En el modo de interfaz gráfica de usuario de la simulación se visualiza en tiempo real como se muestra en la figura 14. En la vista GUI se observa los nodos, lugares, radios de alcance, las rutas actuales, la cantidad de mensajes, etc. Estos son visualizados en todo el campo del mapa, además la velocidad de simulación se puede ajustar con los controles en la parte superior parte de la interfaz gráfica de usuario, y cualquier nodo puede ser seleccionado para

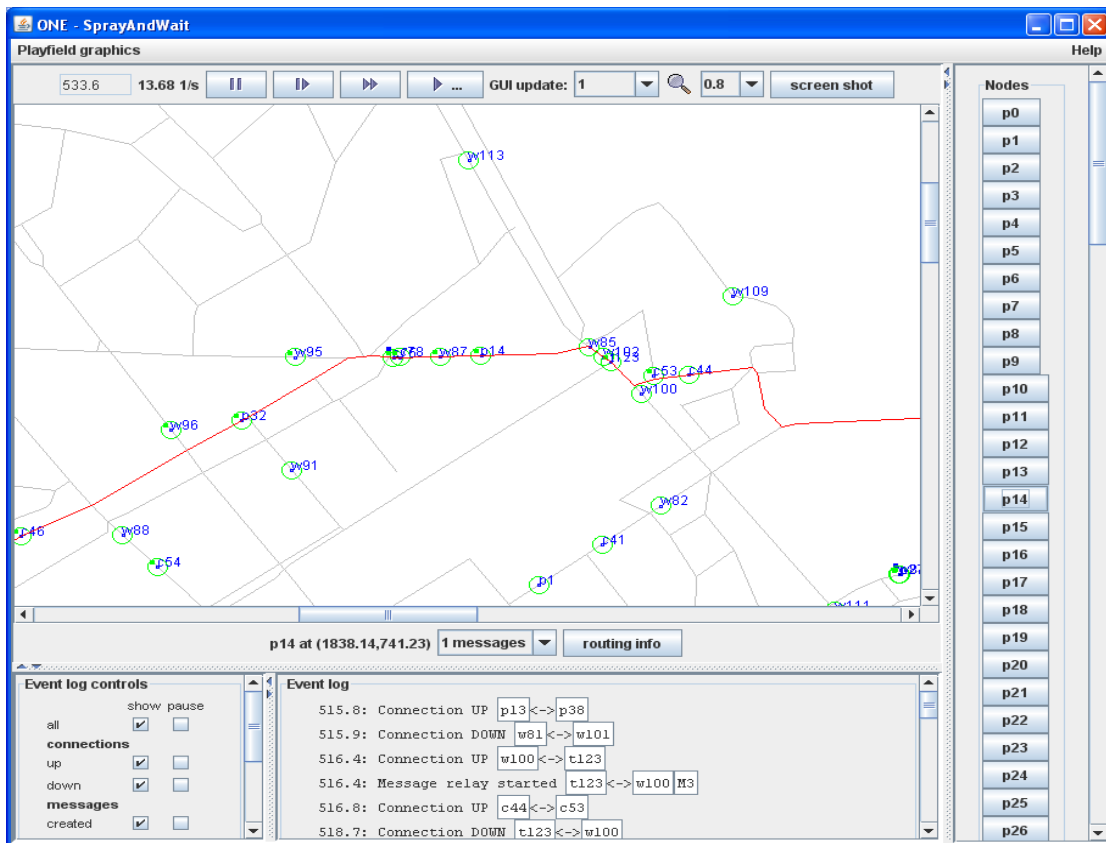


Figura 14 Interfaz de visualización del simulador.

Para la inspección con el panel de selección en el lado derecho. Cuando se selecciona un nodo para la inspección se muestra la ubicación actual y la cantidad de mensajes almacenados provenientes de los distintos contactos realizados con los demás nodos, la simulación se puede detener en cualquier momento con los botones en la parte superior o esperar a que finalice dependiendo del tiempo final configurado en el archivo de configuración default_settings.

Por último, la interfaz gráfica de usuario puede ser útil para fines de demostración, ver el movimiento de los nodos y los mensajes entre ellos, es a menudo más intuitivo que una ventana de la consola la cual solo se puede en cierta manera ver el resultado final cuando la simulación ha terminado.

4.4 Creación de los Mapas

Para crear las rutas por la cuales los diferentes nodos se van a movilizar es necesario generar los mapas, estos van a contener las calles, avenidas, puntos de intereses y demás condiciones para nuestra simulación, para ello es necesario generar un archivo de extensión Well-Known Text WKT, es un tipo de texto marcado como HTML, que se utiliza para guiar las rutas que van a tomar los nodos, para obtener este archivo es necesario un software existen varios uno de ellos es Open Jump⁴, OpenJUMP es un Sistema de Información Geográfica de código abierto (SIG), escrito en el lenguaje de programación Java, es desarrollado y mantenido por un grupo de voluntarios de todo el mundo. Otro software utilizado es osm2wkt⁵, OpenStreetMap es una buena fuente para los datos de los mapas, es gratis y para uso en simulaciones DTN usando The One, osm2wkt convierte archivos de OpenStreetMap lenguaje XML a wkt, además lleva a cabo la conversión de las calles, detecta y elimina partes del mapa particiones y corrige los datos que faltan en algunos puntos.

En la figura 15 se observa la gramática de los archivos wkt, en la cual se definen las coordenadas de conexión de las líneas para las rutas, estas son llamadas "LINESTRING O MULTILINESTRING".

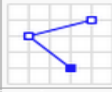
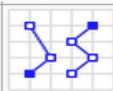
Type	Examples
LineString	<code>LINESTRING (30 10, 10 30, 40 40)</code> 
MultiLineString	<code>MULTILINESTRING ((10 10, 20 20, 10 40), (40 40, 30 30, 40 20, 30 10))</code> 

Figura 15 Ejemplo de creación de líneas, Fuente (36)

El primer punto en la figura muestra el punto de partida el cual es el punto de origen (0,0) está en la posición inferior izquierda. En The One, la

⁴ <http://www.openjump.org/>

⁵ <http://www.tm.kit.edu/~mayer/osm2wkt/>

unidad de la normalización es el metro (m). Ahora en punto (30 10) están expresados en coordenadas (X,Y). Un ejemplo [33], de un archivo wkt es la siguiente cadena: LINESTRING (0 0, 3000 6000, 6000 0, 0 4000, 6000 4000, 0 0, 4000,3000 6000, 6000 4000,6000 0, 0 0), este archivo se encuentra localizado en la carpeta de data con la siguiente dirección “the one 1.4.1/data/customized.wkt”, una vez ejecutado y ajustando otros parámetros⁶ por ejemplo el tamaño del mapa se obtendrá la siguiente grafica 16:

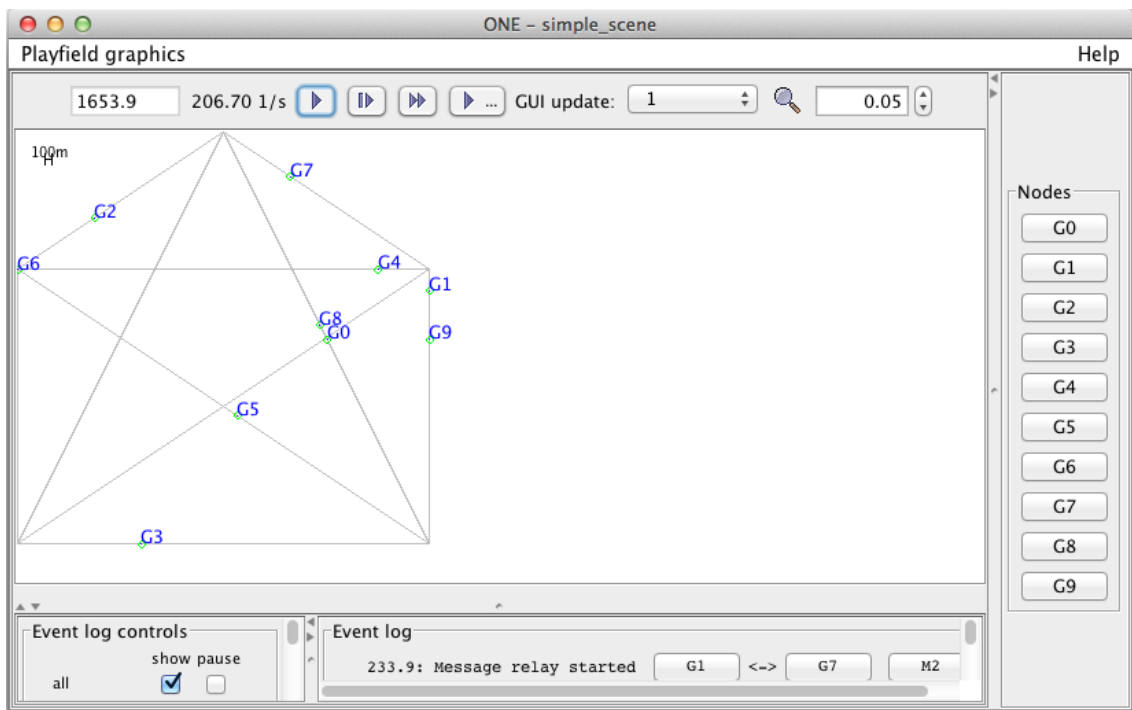


Figura 16 Ejemplo de un ejercicio interfaz final (36)

4.4.1 Definición de escenarios de simulación

Los escenarios de simulación se construyen mediante la definición de los nodos simulados, sus funciones y capacidades. Esto incluye la definición de los parámetros básicos tales como capacidad de almacenamiento, rango de transmisión velocidad, y las tasas de bits, así como la selección del movimiento

⁶ http://www.cs.mun.ca/~xl6488/docs/THE_ONE_003.pdf

específico y modelos de enrutamiento. Otras configuraciones de simulación, como la duración de simulación y tiempos de actualización deben definirse.

El simulador se configura con la configuración basada en texto básico del `default_setings`, archivos que contienen la simulación, interfaz de usuario, la generación de eventos, y los parámetros de la presentación de informes. Todos los módulos tienen su alto nivel comportamiento definido por la aplicación del código Java, pero los detalles del comportamiento se pueden ajustar en el subsistema de configuración. Muchos de los parámetros de simulación se puede configurar por separado para cada grupo de nodos pero los grupos también pueden compartir un conjunto de parámetros y sólo alterar los parámetros que son específicos para el grupo.

The One permite la interacción con módulos externos o la creación de nuevos, los módulos nuevos como plugins se pueden utilizar con diferentes versiones sin necesidad de parchear el resto del simulador. Los Módulos de enrutamiento, modelos de movimiento, generadores de eventos y módulos de informe son todos cargados dinámicamente cuando el simulador se inicia. Además, cuando se crea un nuevo módulo, el usuario sólo tiene que crear y compilar una nueva clase, definir su nombre en la configuración archivo, y el simulador la carga automáticamente cuando el escenario es iniciado. Todos estos módulos también pueden tener cualquier número de ajustes definidos en los archivos de configuración y los ajustes son accesibles para el módulo cuando se carga.

5 Simulación del caso de estudio

En este apartado describiremos la propuesta de escenario de simulación, la cual quiere ser orientada en un entorno de difícil acceso para llevar alguna infraestructura para el funcionamiento de internet. El escenario seleccionado consiste básicamente en 4 nodos estáticos llamados aldeas y un nodo móvil llamado ferry, del cual vamos a analizar los mensajes generados de este hacia los demás nodos.

5.1 Especificación del escenario a simular

Para crear un escenario es necesario tener un mapa de referencia en el cual los nodos van a tomar su movimiento. En nuestro caso, hemos empleado un mapa correspondiente a una región ficticia en la que se ubican cuatro aldeas dispersas, conectadas mediante carreteras. Para simplificar el trabajo, hemos usado como referencia y basado en un mapa del profesor Mauro Margalho Coutinho [37], este se ha creado mediante el software Open Jump del cual ya hablo anteriormente.

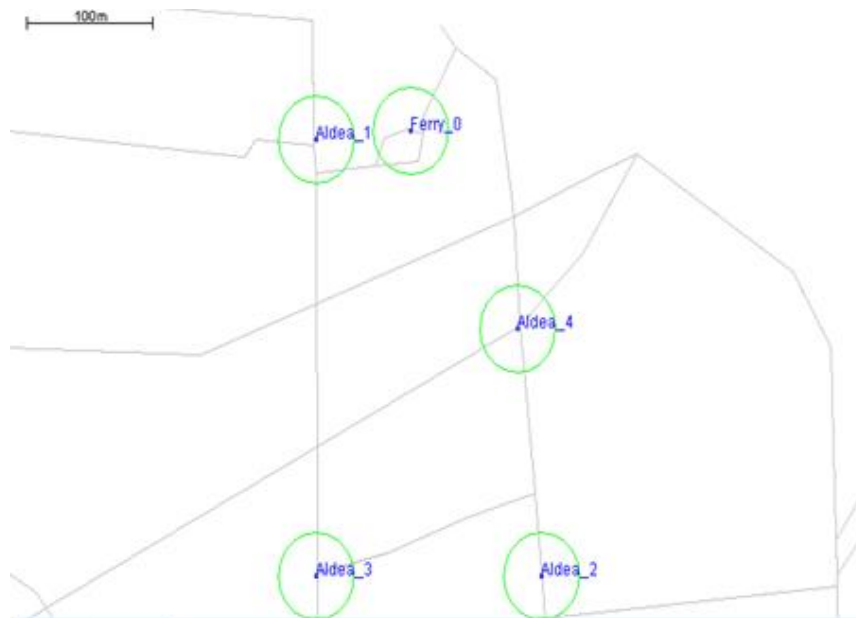


Figura 17 Mapa del escenario simulado

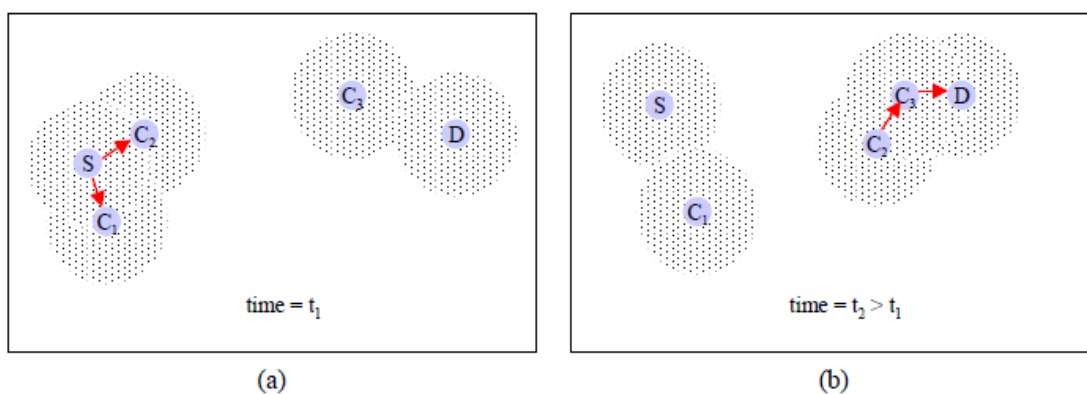
5.1.1 Configuración de los nodos

En nuestro escenario de simulación cada nodo cuenta con una interfaz 802.11b la cual opera a una velocidad de 11Mbps como ya está estipulado en el protocolo IEEE 802.11, los valores deben configurarse en bits es decir que el valor de esta variable es 1375Kbps, otro valor es el rango de transmisión el cual se ha especificado en 10m es suficiente para el modelo de red, otros parámetros se han dejado configurados en su valor por defecto. La tabla 1 muestra los parámetros que consideramos mas relevantes en nuestra simulación, cabe resaltar que algunos varían ya se dijo anteriormente existen 4 nodos estáticos y un nodo móvil, por ultimo se destaca en cuanto a los nodos funcionan como routers indicados en esta sección y próxima.

5.1.2 Configuración de movilidad de los nodos

En nuestro escenario de simulación el único nodo que posee movilidad sigue la ruta definida por el modelo de movimiento llamado: ShortestPathMapBasedMovement, este modelo básicamente utiliza el algoritmo de Dijkstra para encontrar los caminos más cortos entre dos nodos del mapa del azar y Puntos De Interés (POI) como ya se explicó anteriormente en la sección 4.1.1, además existen cuatro nodos estáticos los cuales tienen coordenadas de localización en el mapa, su modelo de movimiento es StationaryMovement en este modelo los nodos no se mueven, es utilizado para simular las aldeas las cuales se encuentran en algún punto fijo dentro de una región.

El protocolo de enrutamiento es otra característica en el escenario para ello se ha elegido EpidemicRouter del cual ya se habló en la sección 3.3.3, para entender un poco mas como funciona en la figura 18 se observa, S desea enviar un mensaje a D, pero hay un camino conectado que está disponible en la parte (a). Los transportistas, C1-C3 se aprovechan para ofrecer transitivamente el mensaje a su destino en algún punto mas adelante en el tiempo como se muestra en (b).



(a) (b)
 Figura 18 Modelo Epidemic Protocol Fuente (26).

5.1.3 Generación de Mensajes

Los mensajes que generan los routers se crean aleatoriamente mediante la clase **Class MessageEventGenerator**, esta crea uniformemente patrones distribuidos de la creación de los mensajes, además los valores de estos mensajes son configurables como el tamaño, intervalo entre la creación de mensajes (segundos), emisor y receptor de los mensajes, es decir si se quiere que el nodo llamado ferry solo envíe mensajes a un receptor a varios, en nuestro caso se ha configurado para que este nodo móvil lleve mensajes a cada los nodos estáticos.

En la siguiente tabla se especifican los parámetros mas relevantes para la configuración de los nodos, en ella se indica para los nodos que son estáticos, para el nodo llamado ferry cambia el modelo de movimiento como ya se indicó anteriormente.

PARAMETRO	VALOR
Tipo de interfaz	Wifi802.11
Velocidad de tx	1375Kbps
Rango de tx	10m
Tamaño del buffer	25M(bytes)
Routing protocol	Epidemic Router
Tiempo min y max de espera	5 , 20 s
Velocidad min y max en trayectoria	0.5 , 1.5
Modelo de movimiento	StationaryMovement
Localización del nodo	429, 400 (x,y)

Tabla 1 parametros de nodos

5.2 Selección de Reports

Básicamente con la simulación se analizaran la información de los mensajes almacenados en cada uno de los nodos estáticos luego de haber transcurrido el tiempo designado para ello, para obtener esta información The One posee varias clases llamadas reports con las cuales generan archivos de texto que van ser almacenados en una carpeta del simulador, se ha configurado ocho clases de las cuales algunas vamos a consultar luego de realizar la simulación estas son:

- **MessageStatsReport:** se utiliza para generar los diferentes tipos de estadísticas sobre el rendimiento total de la transmisión de mensajes.
- **DeliveredMessagesReport:** proporciona la información sobre todos los mensajes enviados.
- **ContactsPerHourReport:** entrega el informe del número de contactos en cada hora.
- **CreatedMessagesReport:** proporciona información acerca de todos los mensajes creados. Se ignoran los mensajes creados durante el período de calentamiento de inicio.
- **DistanceDelayReport:** informa de la distancia de los nodos cuando se envió el mensaje y cuanto tiempo y cuántos saltos tomo en entregarlos. Sólo los mensajes creados después del periodo de calentamiento se cuentan.
- **MessageLocationReport:** informa de la ubicación (coordenadas) de los mensajes, y el reporte de estos.
- **MessageGraphvizReport:** crea una gráfica y genera un informe de esto con el nombre de graphviz, esta grafica contiene los mensajes aprobados.
- **MessageDeliveryReport:** crea un informe para la cantidad de mensajes enviados con respecto al tiempo. Una nueva línea de informe se crea cada vez que se entrega un mensaje.
- **MessageDelayReport:** crea los informes de los retrasos de los mensajes entregados, una línea por cada mensaje enviado y la probabilidad acumulada de entrega por los retrasos en los mensajes.

- **MessageReport:** Informes de los mensajes entregados, tiempo de creación, id del mensaje y tiempo de entrega.
- **EventLogReport:** este es el mismo informe que crea el panel de registro de eventos de la interfaz gráfica de usuario GUI.

6 Resultados de la simulación

En esta sección se expondrán los resultados obtenidos. Como ya explicamos anteriormente, se analizarán los reportes de los mensajes consignados en los informes nos cuales nos indican: efectividad principalmente el protocolo escogido, modelo de movimiento y los parámetros de los nodos como velocidad e interfaz seleccionada que en este caso es Wifi y demás características seleccionadas.

Con los parámetros establecidos procedemos a ejecutar el código, en la figura 19 podemos observar la interfaz grafica de la propuesta de red en The One.

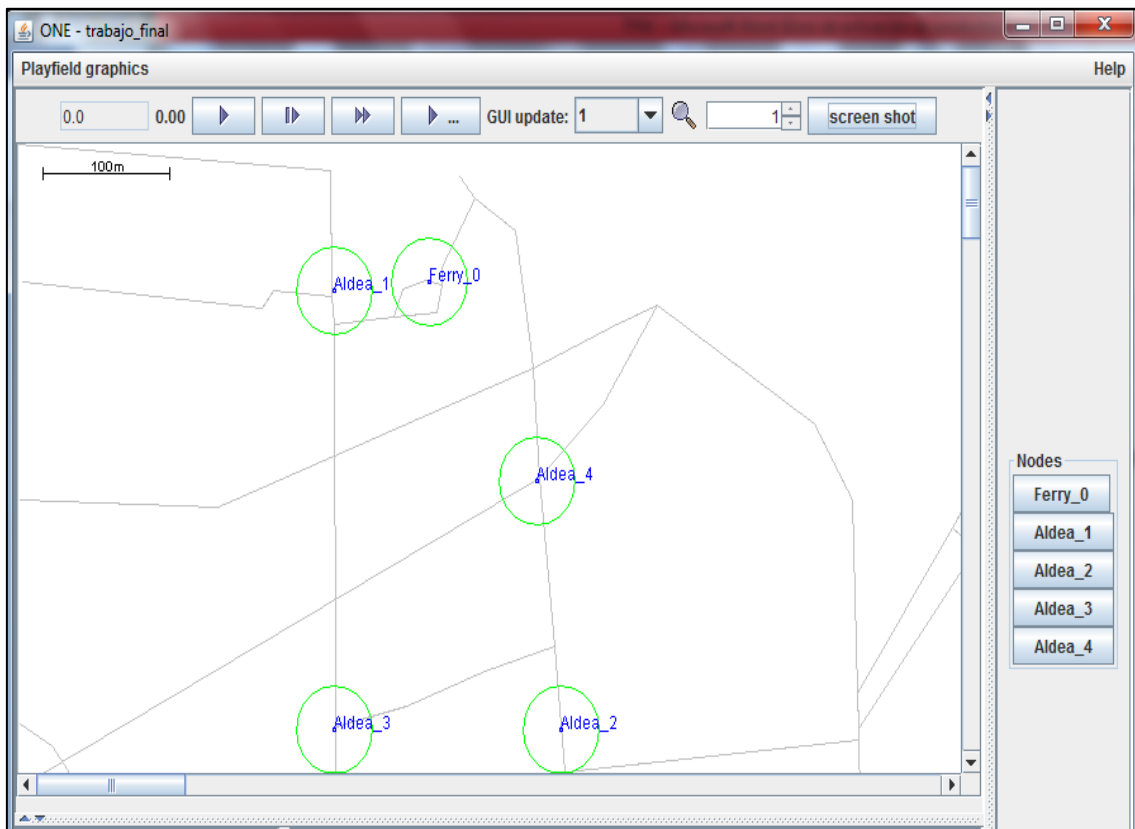


Figura 19 Interfaz Grafica de Propuesta de Red en The One.

Cabe resaltar que los nodos funcionan como router, es decir las aldeas las cuales son el destino de los mensajes también pueden generar mensajes hacia el nodo móvil o hacia otras aldeas, no se logro configurar para solo recibir el mensaje y almacenarlo, luego para obtener los mensajes que solo nos interesan

se han tomados todos los valores y se ha realizado un filtro de solo los mensajes que la fuente es decir el nodo móvil (Ferry) genera hacia el destino es decir las cuatro aldeas.

6.1 Reportes

1. **Created Messages Report:** en el primer informe, la figura 20 muestra los mensajes creados y el tiempos cuando ocurrió ello, los números 1, 2, 3, 4 indican el numero de cada aldea, esto nos ayuda para saber la cantidad de mensajes creados y si necesitamos información en algún tiempo específico, en total se crearon: Aldea_1: 27, Aldea_2: 33, Aldea_3: 27, y finalmente Aldea_4: 18.
- 2.

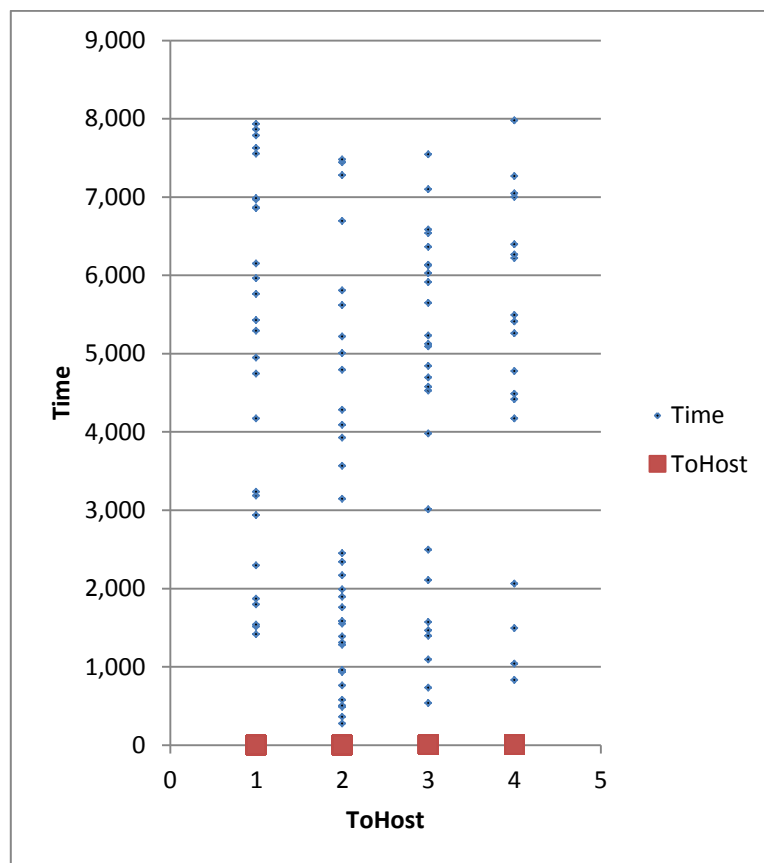


Figura 20 Tiempo de creación mensajes hacia las aldeas.

2. Delivered Messages Report: en la siguiente tabla se observa la cantidad mensajes que finalmente fueron entregados luego de haber sido creados, lo que no importa en realidad es el % de ello, como se puede observar no todos los mensajes fueron entregados a su destinatario las razones pueden ser varias como por ejemplo, el modelo de movilidad, el rango de la interfaz del router móvil, fallas de localización del destino etc.

# Destino Aldea	Mensajes Creados	Mensajes Entregados	%
1	27	0	0,0
2	33	7	21,2
3	27	20	74,1
4	18	11	61,1

Tabla 2 Reporte de Mensajes Entregados

3. Distance Delay Report: este informe nos indica de cuán lejos estaban los nodos cuando se entregó el mensaje, estos valores están en metros (m), en caso tal que no se haya logrado entregar un mensaje el tiempo de entrega(segunda columna) y el número de saltos (cuarta columna) se ponen en (-1), para el ejemplo se resalta, el mensaje con ID A6 fue creado en el segundo 80 como se muestra en la figura 21 está dirigido hacia la Aldea_1 como podemos observar en este caso el tiempo de entrega del mensaje y numero de saltos esta en (-1) ya que no se logro entregar el mensaje.

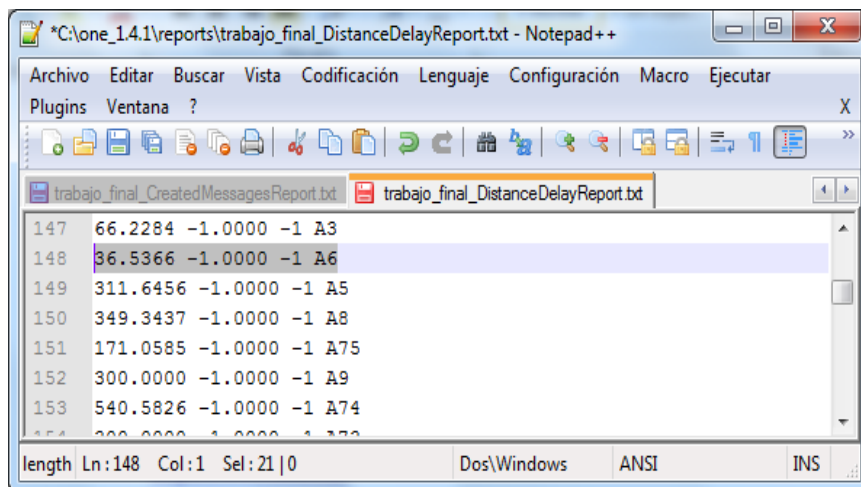


Figura 21 Reporte DistanceDelayReport

3. **Message Delay Report:** este reporte obtenemos los retardos de los mensajes (una línea por mensaje entregado), estos son comparados con el reporte llamado DeliveredMessagesReport, este tiempo se mide desde que los datos fueron generados hasta que finalmente fueron entregados, además nos indica una probabilidad acumulativa de entrega. Esta nos ayuda si necesitamos para saber la efectividad de los parámetros de configuración establecidos, en nuestro caso como ya se observó anteriormente con el reporte de DeliveredMessagesReport no fue muy eficaz ya que se perdieron todos los datos generados hacia la Aldea_1.

	messageDelay	cumulativeProbability
1		
2	0.7000	0.0018
3	0.8000	0.0036
4	10.4000	0.0054
5	27.6000	0.0072
6	35.2000	0.0091
7	46.3000	0.0109
8	53.9000	0.0127
9	54.0000	0.0145
10	82.3000	0.0163
11	105.2000	0.0181
12	105.3000	0.0199
13	110.0000	0.0217
14	113.6000	0.0236
15	118.7000	0.0254
16	137.3000	0.0272

Figura 22 Reporte MessageDelayReport

5. **MessageStatsReport:** Finalmente obtenemos el último reporte con este se obtiene los diferentes tipos de estadísticas sobre el rendimiento del total de la transmisión de mensajes, allí vemos que los diferentes valores a los inicialmente mostrados ya que como se dijo anteriormente los nodos funcionan como router y generan mensajes a todos los demás nodos.

The image shows a Notepad++ window with the following content:

```
1 Message stats for scenario trabajo_final
2 sim_time: 8000.0000
3 created: 552
4 started: 338
5 relayed: 328
6 aborted: 10
7 dropped: 622
8 removed: 0
9 delivered: 94
10 delivery_prob: 0.1703
11 response_prob: 0.0000
12 overhead_ratio: 2.4894
13 latency_avg: 920.9394
14 latency_med: 699.4000
15 hopcount_avg: 1.2447
16 hopcount_med: 1
17 buffertime_avg: 1574.3920
18 buffertime_med: 1730.5000
19 rtt_avg: NaN
20 rtt_med: NaN
21
```

The status bar at the bottom indicates: length Ln:9 Col:14 Sel:0|0, encoding: UNIX, font: ANSI, and style: INS.

Figura 23 Reporte MessageStatusReport

7 Conclusiones

A lo largo de esta tesina y a tenor de los resultados expuestos en el capítulo anterior creemos que ha quedado demostrado la importancia del estudio de las redes DTN en ambientes de difícil acceso para instalar una infraestructura para funcionamiento de internet, además de ellos la simulación nos da bastaste una perspectiva si se desea instalar o realizar pruebas de ello, los resultados obtenidos nos ayudan a tomar dediciones si los parámetros escogidos para la simulación son suficientes o eficaces para nuestro objetivo, en nuestro caso podemos observar que es necesario variar o realizar pruebas con otro tipo de enrutamiento y modelos de movilidad y la configuración nodos para que funcionen como solo destino y no como router, ya que de un total de 338 mensajes enviados obtuvimos lograron llegar a su destino 94 es decir el 17%, como se ve en la figura también influye en ello el rango de la interfaz de los router que en nuestro caso ha sido de 10m y con el protocolo escogido 802.11 es posible aumentar este rango decidimos escoger un mínimo rango para obtener tal funcionalidad.

Por últimos somos conscientes de que hay que realizar más simulaciones o aumentar el tiempo de ello, además poder comparar con otros trabajos realizados otros instituciones, se consultó pero las simulaciones que se consultaron ninguna tenía el modelo propuesto por nosotros.

8 Trabajo Futuro

Una vez finalizado el desarrollo del proyecto, es conveniente considerar nuevas características que puedan mejorar los objetivos propuestos, a continuación brevemente se exponen ideas para en un futuro el desarrollo de ello:

- Mejorar características para el modelo de solución: Una de las acciones mas importantes es el diseño de modelo definido para el router móvil, es decir que siga una ruta establecida con el fin de realizar en contacto con los otros nodos, además lograr que los nodos estáticos en este nuestro trabajo llamado aldeas funcionen como solo destino y no como router, con el fin que estos no generen mensajes si no solo sean receptores de estos.
- Explorar nuevas propuestas: Ya con el tema propuesto y el estudio realizado de las redes DTN sería conveniente poder realizar más

simulaciones con otros protocolos y realizan comparaciones si es posible encontrar investigaciones del mismo tema con el fin de obtener nuevas ideas para su mejora.

Bibliografía

[1] La NASA prueba con éxito el DTN o versión del IP para aplicaciones espaciales, pag web.

http://www.nasa.gov/home/hqnews/2012/nov/HQ_12-391_DTN.html

[2] Networking for Communications Challenged Communities: Architecture, Test Beds and Innovative Alliances, pag web <http://www.n4c.eu/>

[3] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss. Delay-Tolerant Networking Architecture. Number RFC 4838. IETF, April 2007.

[4] SCOTT, K., AND BURLEIGH, S. Bundle Protocol Specification. RFC 5050, November 2007.

[5] A. Seth, D. Kroeker, M. Zaharia, S. Guo, S. Keshav, Low-cost communication for rural internet kiosks using mechanical backhaul, in: MobiCom'06: Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, ACM Press, New York, NY, USA, 2006, pp. 334-345.

[6] S. Guo, M. Derakhshani, M. H. Falaki, U. Ismail, R. Luk, E. A. Oliver, S. U. Rahman, A. Seth, M. A. Zaharia, and S. Keshav, "Design and implementation of the KioskNet system," *Comput. Netw.*, vol. 55, no. 1, Jan. 2011, pp. 264-281

[7] Rede para Comunicação em Comunidades Remotas - N4C (www.n4c.eu)

[8] Disruption Tolerant Networking for Space Operations (DTN), pag web http://www.nasa.gov/mission_pages/station/research/experiments/730.html

[9] S. Parikh and R. Durst, "Disruption tolerant networking for marine corps CONDOR," in Military Communications Conference, 2005.MILCOM 2005. IEEE, vol. 1, Oct. 2005, pp. 325-330.

[10] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet," *SIGOPS Oper. Syst. Rev.*, vol. 36, no. 5, Oct. 2002, pp. 96-107.

[11] <http://www.ibr.cs.tu-bs.de/projects/emma/>

[12] P. Bonnet and M. Chang, "Monitoring in a high-arctic environment: Some lessons from mana," *IEE Pervasive Computing*, vol. 9, no. 4, Oct. 2010, pp. 16-23.

[13] K. Römer, O. Kasten, y F. Mattern. Middleware Challenges for Wireless Sensor Networks. *SIGMOBILE: Mobile Computing and Communications Review*, 6(4):59-61, Octobre 2002.

[14] S. Domancich, "Security in delay tolerant networks for the Android platform," Master's thesis, Royal Institute of Technology (KTH), Sweden, June 2010

[15] O. Mukhtar and J. Ott, "Backup and bypass: Introducing DTN-based ad-hoc networking to mobile phones," in *Proceedings of the 2nd International Workshop on Multi-hop Ad hoc Networks: From Theory to Reality (REALMAN '06)*. New York, NY, USA: ACM, 2006, pp. 107-109

[16] <http://www.dtnrg.org/wiki/Code>

[17] <http://www.dtnrg.org/wiki/Home>

[18] <http://www.ietf.org/rfc/rfc5326.txt>

[19] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, E. Travis, and H. Weiss. *Delay-Tolerant Network Architecture: The Evolving Interplanetary Internet*. IPN Research Group, 2002.

[20] Berners-Lee, T., Fielding, R., and Masinter, L., —Uniform Resource Identifier (URI): Generic Syntax, Internet RFC 3986, January 2005.

[21] F. Warthman, *Delay-Tolerant Network, A Tutorial*, Mayo 2003.

[22] L. Wood, W. Eddy, and P. Holliday, "A bundle of problems," in 2009 IEEE Aerospace Conference Proceedings. Piscataway, NJ, USA: IEEE, 2009

- [23] N. Bezirgiannidis and V. Tsaoussidis, "Packet size and DTN transport service: Evaluation on a DTN testbed," in Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2010 International Congress on. IEEE, 2010, pp. 1198-1205.
- [24] Jain, S., Fall, K., and Patra, R., —Routing in a Delay Tolerant Network, SIGCOMM Comput. Commun. Rev. Vol.34, Iss.4, p145-158, August 2004.
- [25] A. Lindgren and A. Doria, Probabilistic Routing Protocol for Intermittently Connected Networks, Internet Draft, Aug. 2010. [Online].Available: <http://tools.ietf.org/html/draft-irtf-dtnrg-prophet>.
- [26] A. Vahdat and D. Becker, BEpidemic routing for partially-connected ad hoc networks, Duke Tech. Rep. CS-2000-06, Jul. 2000.
- [27] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, BSpray and wait: An efficient routing scheme for intermittently connected mobile networks,[in Proc. ACM SIGCOMM Workshop Delay-Tolerant Netw., 2005, pp. 252-259.
- [28] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, BMaxProp: Routing for veicle-based disruption-tolerant networks, in Proc. IEEE INFOCOM, Barcelona, Spain, DOI: 10.1109/INFOCOM.2006.228.
- [29] R. Frank, E. Giordano, P. Cataldi, and M. Gerla. TrafRoute: A diferent approach to routing in vehicular networks. In Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on, pages 521{528. IEEE, October 2010.
- [30] <http://watwire.uwaterloo.ca/DTN/sim/>
- [31] <http://www.isi.edu/nsnam/ns/>
- [32] <http://www.omnetpp.org/>
- [33] http://www.cs.mun.ca/~xl6488/docs/THE_ONE_003.pdf
- [34] paper Ari Keränen, Jörg Ott, Teemu Kärkkäinen , The ONE Simulator for DTN Protocol Evaluation, Helsinki University of Technology (TKK) http://www.netlab.tkk.fi/tutkimus/dtn/theone/pub/the_one_simutools.pdf

[35] A. Keranen, Opportunistic Network Environment simulator, Helsinki University of Technology Department of Communications and Networking Special assignment, May 29, 2008.

[36] Xu Liu _and Yuanzhu Chen THE ONE - Case2: Customizing The Movement of Nodes, May 11, 2013

[37] Mauro Margalho Coutinho, The One Tutorial, www.margalho.pro.br