Advances in Wireless Community Networks with the Community-Lab Testbed

Leandro Navarro (UPC), Roger Baig Viñas (guifi.net), Christoph Barz (Fraunhofer FKIE), Joseph Bonicioli (AWMN), Bart Braem (iMinds), Felix Freitag (UPC), Ivan Vilata-i-Balaguer (Pangea)

Abstract:
Beyond traditional telecom providers, citizens and organizations pool their own resources and coordinate in order to build local network infrastructures to address the digital divide in many parts of the world. These crowdsourced network infrastructures can be self-organized and shared by a community for the collective benefit of its members. Several of these networks have developed open, free, and neutral agreements, and are governed as a common-pool resource: Community networks. These are built using a variety of commodity wireless hardware (e.g. Wi-Fi long range point to point links, Wi-Fi and GSM access points, and mesh networks), sometimes optical fibre links, heterogeneous nodes, routing protocols and applications. A group of researchers, developers and community networks developed the Community-Lab testbed, and for the last five years have worked together to overcome obstacles, improve the technologies, tools and operational models being used, as well as model best practices for more effective and sustainable community networks. This article presents the challenges for experimentation, the testbeds built, results, lessons learned and the impact of that work to place wireless community networks as one sustainable way towards an Internet accessible to All.

Introduction

Access to the Internet is increasingly seen as a fundamental opportunity in allowing individuals to participate in society and benefit from services relating to all aspects of our lives. Despite the fact that connectivity is also considered a key component for socio-economic development all over the world, there is a digital divide. The Global Internet Report (Internet Society) estimates that there were 3 billion people online in 2015 out of a population exceeding 7 billion. However, alternatives and choice of connectivity services are considered a necessity for accessibility (based on cost and quality) and sustainability (to avoid dependency).

Traditionally, this access has been mainly offered by commercial ISPs and telecom providers. However, their economic model has left a large fraction of the population behind, such as inhabitants of remote areas, with low population density and high deployment costs, or people of low economic capacity, or those simply not satisfied with the local commercial offerings.

Beyond traditional telecom providers, there is an alternative method with which to address the digital divide. Citizens and organizations can pool their own resources and coordinate in order to build local network infrastructures. These crowdsourced network infrastructures can be self-organized and shared by a community for the collective benefit of its members.

These models of participation, organization, and financing of local network infrastructures can vary widely. For example, some networks are freely accessible, others are cooperation-based and some are run by federations of microISPs, etc. The following examples demonstrate the diversity described:

Broadband for Rural North (B4RN), in Lancashire (UK), deploys and operates optical fibre in a cooperative way. The Nepal Wireless Networking Project (NWNP) is a social enterprise that, due to the lack of alternatives, provides Internet access, e-commerce, education, telemedicine, environmental and agricultural services to many remote villages, using wireless technologies. The French Data Network Federation (FFDN) is a federation of French Do-it-Yourself ISPs, which comprises DSL resellers, wireless Internet service providers (WISP), colocation centres, and other similar organizations. Village Telco is a model of wireless mesh networks replicated in several communities around the world, such as the Zenzeleni rural telecom cooperative, the first legally established in rural South-Africa in 2015. Rhizomatica, in Mexico, has deployed open-source GSM base stations [12] in more than 10 rural communities under a new national spectrum regulation scheme for rural and indigenous areas. The Internet Society has supported several pilots of Wi-Fi-based networks deployed in remote areas of India by locals [5].

Several crowdsourced networks have developed open, free, and neutral agreements or licenses in a similar way to the Free and Open Source Software model [2]. Openness suggests that everyone has the right to know how they are built. Freedom indicates that the network access is driven by a non-discriminatory principle, and therefore, they are universal. Finally, neutrality implies that any technical solution available could be used to extend the network, and that it should be used without discrimination. This means that the network can be used to transmit data of any kind by any participant, including for commercial purposes. They are so called Community Networks (CN). Representative community network examples are: Freifunk (FF) in Germany, the Athens Wireless Metropolitan Network (AWMN) in Greece, FunkFeuer (0xFF) in Austria, guifi.net in Spain, and Ninux.org in Italy.

Community networks such as the ones presented above, are built using a variety of commodity wireless hardware (e.g. Wi-Fi long range point to point links, Wi-Fi and GSM access points, and mesh networks), sometimes optical fibre links, heterogeneous nodes, routing protocols and applications. They support a large number of end users (guifi.net reaches over 30,000), and follow an innovative model of self-provisioning using an unlicensed and public spectrum [4]. These infrastructures, having been developed cooperatively for the common interest and become a key resource for local communities, a networking community where digital social life takes place. Individual citizens, public institutions, NGOs, businesses and service providers themselves contribute and take advantage of this common infrastructure. These community networks are an emerging and successful model for the Future Internet, with hundreds of examples around the world.

Aiming to support community network infrastructures, a group of research and community organizations developed the Community-Lab testbed. This testbed was launched with the support of the CONFINE Future Internet Research and Experimentation project (2011-2015), supported by the European Commission

as one of its Future Internet testbeds, and federated with other global testbeds (Fed4FIRE and GENI). The project and the testbed include well established community networks, with large end-user bases and diverse application providers, research institutions, non-profit organizations and small enterprises. The team comes from 24 partner organizations from Europe, America and Africa.

In 2016 the Community-Lab testbed has more than 200 hosts embedded in several community networks, and is deployed within the federation of existing CNs comprising more than 40,000 routers and 60,000 Km of links. Community-Lab offers unified access to the testbed, with tools that allow researchers to deploy, run, monitor and experiment with services, protocols and applications on real-world CNs. Experimenting in production networks is challenging, as previous experiences such as Planetlab [6] demonstrate. For the last five years communities of practitioners have worked together with researchers and developers to overcome obstacles, improve the technologies, tools and operational models being used, as well as model best practices for more effective and sustainable community networks. This article presents the challenges for experimentation, the testbeds built, the results, lessons learned and the impact of the research to advance wireless community networks as one sustainable way towards an Internet accessible to All.

Challenges and Issues

Community networks face many challenges [9]. For instance, many of the technologies used were not designed or optimized for this particular purpose. As a result, local infrastructures can be very fragile. Moreover, self-organization and regulation are necessary in order to avoid abuse and collapse, which involves extra effort. These goals should be achieved by creating an environment able to accommodate many people in a sustainable manner.

The active involvement of end users in these networks, usually very skilled, curious and with an entrepreneurial attitude in trying and adopting new solutions even in prototype form, offers a unique opportunity for discussing and suggesting ideas and improvements. In several situations, for instance, routing issues such as loops or scalability problems, the research community has been successful in characterizing problems (e.g. from logs and benchmarking) and contributing solutions (e.g. deploying experimental routing extensions). However, collaborations require intensive effort from practitioners and researchers. The existence of a testbed has proved to facilitate, automate and systematize the collection of information (key to understanding and formalizing research problems) and the execution of experiments under realistic conditions and different scenarios.

The aim of the Community-Lab testbed, is to advance research and empower society, by understanding and removing obstacles to the adoption of CNs and their services. The main objective is to support experimentally driven research, improving software and network development as well as the evaluation of the multiple aspects of community networks. Community-Lab also provides a set of hosts (embedded in several community networks) that are specifically adapted to running experiments, and which allow the involvement of end users. The

main features of the testbed can be summarized using the following adjectives: *realistic*, *heterogeneous*, *integrated*, *open*, and *participatory*.

The testbed is *realistic* because it functions under the operational conditions of several community networks and allows the identification of potential issues in the development and operation of new services.

The *heterogeneity* of the testbed is the result of providing access to very diverse environments in terms of node and link characteristics, availability and reliability, among other aspects.

The *integration* of experimental nodes and services in order to run experiments within a common framework, with a common web portal (community-lab.net), saves time and costs for the users. This simplifies the life cycle of experiments and pilot testing, and allows the porting of experiments across other federated experimentation testbeds.

The *openness* of the testbed facilitates the participation of diverse research groups and CNs, for both short term and long term experimentation. Moreover, service providers and entrepreneurs can use the testbed to try out innovative services. In addition, openness facilitates verification and repeatability of the experimental conditions and results since the infrastructure is always available.

The main challenges in the implementation of testbed was the design and implementation of the following mechanisms and technologies:

- Resource sharing is critical, to allow multiple concurrent experiments, but isolation is an issue.
- Cross-layer solutions are needed in order to optimize services such as routing (with development and experiments of mesh routing algorithms such as BMX6 or OLSR2), content distribution, peer-to-peer live video streaming, information-centric networking, aggregation and sharing of Internet access capacity [11].
- Self-managing systems running in large and dynamic networks, self-configuration (experiments in adaptive radio channel allocation, power and rate), self-healing (experiments with services that adapt to node or link failures), self-optimization (adaptation according to available resources and depending on internal or external influences).
- Availability of open data sets such as routing graphs, traffic maps and traces, topology change dynamics, load, failures, number of users, volume of traffic and spectrum occupation.
- Development of a benchmarking framework, enabling experiments under controlled and reproducible test conditions, offering automated procedures for experiments and performance evaluation and allowing a fair comparison between different networking mechanisms.
- Technical adoption, evaluation of how to incorporate the research results into the hardware and software infrastructure of existing communities and its application to other areas.
- Social mechanisms allowing the assessment of the social impact of the development of these networks on the future Internet and on the inclusion of more people in the digital society.
- Economic mechanisms in order to explore the conditions for the sustainability of these infrastructures, influence of cooperative schemes

for infrastructure sharing, financing and the provision of services such as a community cloud, to end users and local communities.
- Legal mechanisms that would enable us to explore the feasibility of these models, the role of governments in the regulation of the use of public space, spectrum, and provision of services.

Community-Lab testbeds

The set of facilities for experimentation developed in the CONFINE project has the collective name of Community-Lab. A key aspect considered in the design of the Community-Lab was the fact that experimentation can have a strong impact on production networks with limited resources, and therefore, these networks would need to be protected. In that sense, we realized that experimentation can benefit from different testbeds specifically optimized to give control to the users over some aspects while providing an abstraction level for others.

The facilities in Community-Lab are shown as columns in Figure 1. The **Community Network testbed** deployed inside several community networks, the **Virtual testbed** that can be instantiated within a single computer, and the **Campus testbed** that can be deployed in specific areas, temporarily or permanently.
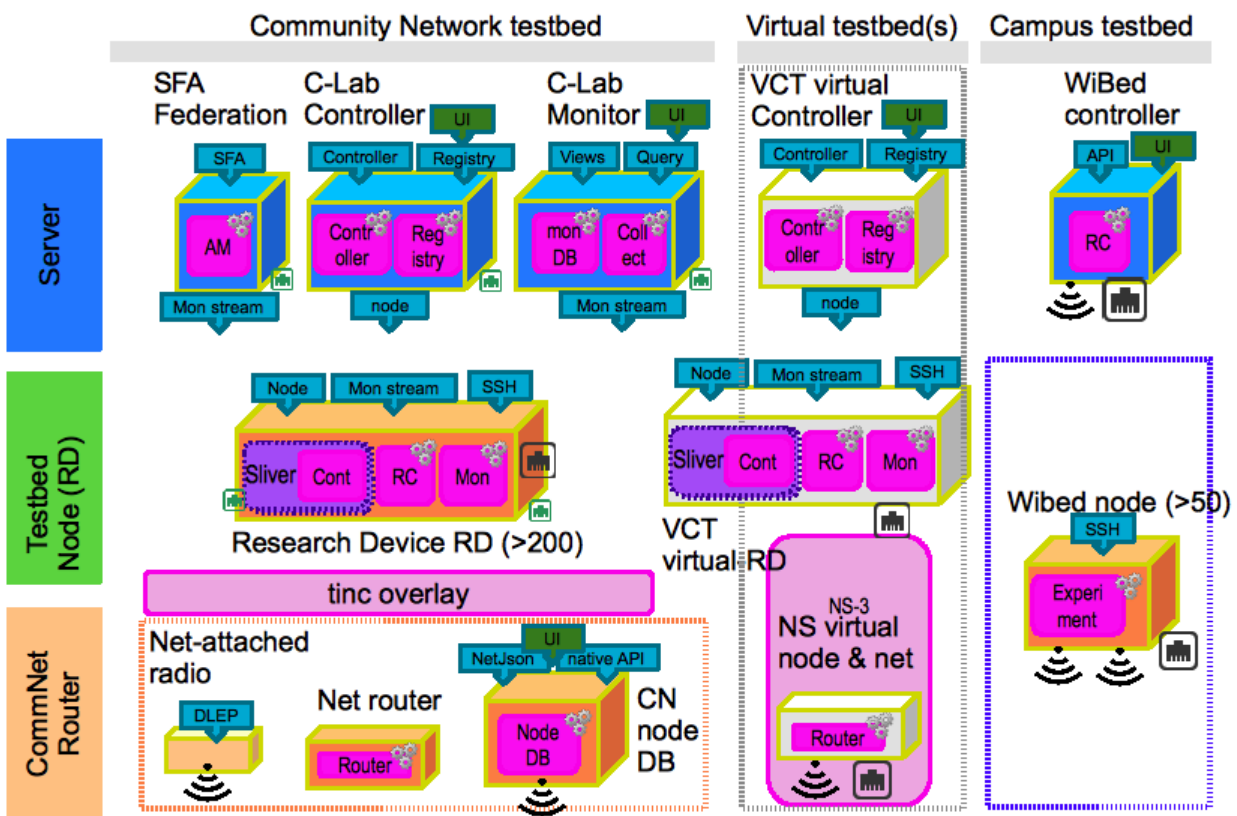

Figure 1: The node/functional architecture of the Community-Lab (C-Lab) testbed

Each of these experimentation environments can be divided in three groups of components (rows in Figure 1): *servers and services* controlling the testbed, *testbed nodes (i.e. research devices),* where experiments are performed, and network routers, which are part of community networks.

The largest, widest and publicly accessible testbed (for registered researchers) in Community-Lab is the **Community Network testbed** (first column from the left in Figure 1) with testbed nodes deployed as hosts embedded in several production community networks (guifi.net, AWMN, FunkFeuer, FKIE, Wireless België, Ninux, Sarantaporo.gr), with a few more community networks involved in social experiments and development (Figure 2). This testbed has three main services: a) the Federation service that interconnects with other testbeds in the Fed4FIRE federation, b) the Controller service that allows the experimenter to define slices and select nodes (RD) in order to deploy sliver templates, informs each RD about its desired state (slivers requested for a given RD), and collects and aggregates sliver status information, and c) the Monitor that collects, aggregates and presents monitoring information from each RD connected to the underlying CN (this can be seen as an IP network of interconnected routers).

We used Tinc, an IPv6 overlay management network with multiple gateways, to offer a common bidirectional network on top of either IPv4 or IPv6 networks, NAT or filters, and across Europe using tunnels over the GÉANT academic network and the Internet [1]. This overlay offered a seamless management network able to reach any testbed host, efficient in optimizing routes across gateways and nodes, and protected from the open Internet by a public key mechanism. Following the Planetlab slice-base architecture, an experiment controller allows the selection of which nodes (Research Devices) to use for an experiment. Each node checks with the controller and creates any required sliver (Linux container).
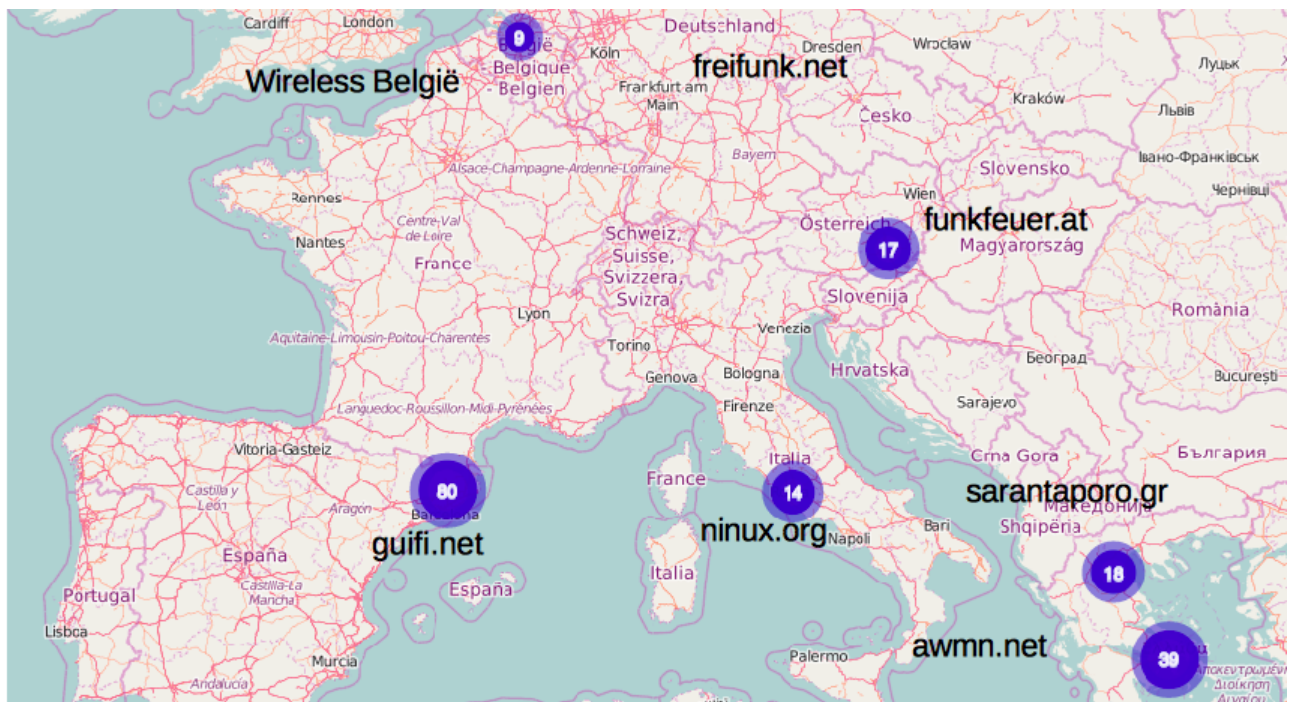
Figure 2. The Community-Lab Community Network testbed (2015).

The coexistence with production networks is a key differentiating factor for Community-Lab, but it depends on the enforcement of limitations on uncontrolled experiments that may otherwise stress the underlying network in terms of traffic and low-level network access, and cause interference, intrusions or leakage of private data. With that goal in mind, the **Virtual testbed** can be used to test experiments during development, test the testbed software itself before deployment in production networks, or run a very intrusive experiment using the same set of tools as in the real testbed.

The **Virtual testbed** is a software package which allows the deployment of a complete testbed in a single computer (or a cluster), using virtual machines and a virtual network. This can also simulate virtual wireless links through the integration of the ns-3 network simulator. It is depicted in the central column in Figure 1. An instance of a virtual testbed should contain one controller, a set of virtual RDs, a virtual network topology, and a virtual network. Since the software is nearly the same as in the main testbed and there is also an underlying simulated network, an experiment can be easily ported from the virtual to real testbeds.

The **Campus testbed**, also known as WiBed [3], was developed in order to avoid the network-protecting limitations set by the community network testbed. It has a specific architecture which allows radio and router level experiments in a separate network for potentially network-disturbing and radio-interfering experiments, that may not be compatible with a production CN. It is depicted in the first column from the right in Figure 1. The WiBed testbed nodes can run experimental firmware images that are downloaded from a control server, the WiBed controller, which provides a Web and API interfaces to control nodes and experiments. The most typical experiment deployed on the Campus testbed has been the testing and performance characterization of mesh routing protocols.

There are several instances of this testbed; Around 40 router-like devices are deployed at different offices in several buildings at the UPC. These router devices have three radios. Two of them also include an Ethernet connection for control. As WiBed is both a specific deployment and a software package, it can also be deployed in other locations. This has been done during BattleMesh workshops in Leipzig, Germany (2014), in Maribor, Slovenia (2015), and in Porto, Portugal (2016).

The Community-Lab.net website describes how researchers can register to propose an experiment that uses the community network testbed, or can simply learn how to deploy the community network or the campus testbed in a new location, or the virtual testbed on a PC. Community networks can also find instructions in order to join the testbed.

Results, Lessons and Impact

The main results of this collaboration between communities, researchers and developers in the last five years can be broadly divided into three areas. A first set of results is derived from the **testbed software and experimentation services** themselves. The second set concerns **experimental research**. The

third set focuses on the development of **business models, systems, and services**, applied or applicable to community networks in general.

Result	Use / Impact
Testbed Software and Experimentation Services	
C-Lab system	Experiment controller, API, Confine Node System (CNS)
C-Lab deployment	200+ Worker nodes (Research Devices) and servers
C-Lab VCT	Software product, integration
C-Lab tools	Monitor, Federation of testbeds and experimental service clouds
C-Lab WiBed	Software product, documentation, experiments
Results from Experimental Research	
OLSRv2	Algorithms, Reference implementation, input to standardization
DLEP	Reference implementation, input to standardization
BMX6	Algorithms, software product, research pubs
NodeDB (netJSON)	API, reference implementation
Cloudy Community Cloud	Software product, integration, experimental use
Open Data-Sets	Topology, traffic, routing, configuration, participation
Knowledge on computer networks	Routing algorithms: evaluation and evolution, resource allocation algorithms, cross-layer optimizations
Knowledge on systems	Scalable systems, interference, resilient systems
Business Models, Systems, and Services	
Knowledge on communities	Organizational models, Common-Pool Resource model, Structural and topologic properties

Table 1. Summary of the main results

Testbed Software and Experimentation Services

The main available assets for experimentation are the software tools to build similar testbeds, the services for experimentation, and several open-data sets.

Testbed Software:
Several software tools for large-scale experimentation, under an Open Source/Free Software model. (Software repository: http://redmine.confine-project.eu and documentation: http://wiki.confine-project.eu). These software tools allow extending or building a separate testbed elsewhere.

Services for Experimentation:
Community-Lab provides three environments in which to perform experiments in a set of nodes and networks (a slice):
- The Community Network testbed, to deploy experiments embedded in several production community networks. It can be accessed through its Community-lab.net web portal or through the Fed4FIRE SFA federation.

- The virtual testbed that is deployed on a single computer and can emulate a network testbed for development or testing purposes.
- The campus testbed (WiBed), deployed in a campus environment.

Open-data sets:
The participating community networks and research activities have created open-data sets (http://opendata.confine-project.eu) in open-access mode, already used in several published research experiments. Some data sets are generic, such as network graphs, traffic or participation. Other data sets are used in specific publications which are made available to enable others to reproduce, verify, or extend the results or simply use them in their own research activities.

Experimental Research

The main experimental results are the following:

Dynamic Link Exchange Protocol (DLEP)

DLEP is a protocol that standardizes a local connection between a bridging radio and a router, to allow the router to learn about the available link-layer and physical layer data (e.g. bitrate, frequency or signal strength). All control data is sent via the same interface where the user data traffic is sent. Our implementation for OpenWRT included additional physical layer data such as signal strength, frequency or bitrate, and additional link-layer data, mainly including statistics such as number of sent and received frames, lost frames and retransmissions. Research Devices and community network routers can use this information for improved routing metrics, network measurements or application-layer cross optimizations. [15]

OLSRv2

We improved and extended the second generation *OLSRv2* implementation for OpenWRT Linux and tested the protocol in the virtual and campus testbeds (Battlemesh 2015). We have also developed and evaluated *multi-topology extensions* and the directional airtime metric for OLSRv2 experimentally. While most community networks do not integrate the concept of priority traffic into their networking mechanisms, to retain fairness among users, there are reasons why some traffic could be prioritized or restricted to a subset of links. Such cases as certain real-time applications, for example voice over IP or video communication that need reliable low-delay links, which, in contrast to BitTorrent, are more robust against delay and packet loss. [15]

Receiver-Driven routing in BMX6

A desirable feature for community network routing protocols is the incorporation of a list of preferred receiver-nodes in the routing algorithm, thus enabling routing algorithms to respect the community social contract and not restrict the freedom of community users. In our receiver-driven discretionary routing mechanism, each receiver (the intended destination of the packet) can freely specify delivery objectives and remain compatible with the collaborative approach of community networks. Each node has a unique identifier and can announce the description of its offer and also the description of its routing policy with preferences to deliver traffic to that node. This receiver-driven routing can be applied to express preferences for desirable nodes and paths, or to restrict traffic to trusted nodes enabling trust and security aware routing. A

proof-of-concept implementation of key concepts, developed as an extension of the BMX6 routing protocol, validated in our virtual and campus testbeds, confirms its feasibility and scalability [10]. This work also allows a routing protocol to perform cryptographically secured negotiation and establishment of concurrent and individually-trusted routing topologies for infrastructure-less networks without relying on any central management. Finally, this feature can help in protecting a community network from problems which result from the trustworthiness of participants within an increasing number of users and diversity.

NodeDB (netJSON)

For community wireless networks (CWN), a node database serves as a central repository of network information. This "registry" functionality is separate from the testbed controller, described above. The common NodeDB manages the network information per se, for the community network. NodeDB is simultaneously a registry, a link planning tool, an IP address assignment tool, etc. The information stored concerns nodes deployed at certain locations, devices installed at these locations, information regarding internet addresses, and for networks that use explicit link planning, links among devices. All this information is maintained via a web or RESTful interface by the community members. Therefore, the common NodeDB contains the static as well as dynamic information about each *community network* as opposed to the experimental testbed network information.

Community Clouds (Cloudy)

Cloudy is a Linux distribution designed for building community cloud infrastructures in community networks [14]. The distribution contains the platform and application services of the community cloud model. It has been tested in the Community-Lab testbed and then deployed across approximately 70 Atom nodes, most of which are located in community members' homes. It has a decentralized discovery service (Serf based), and a small set of services, including social interaction, video streaming, file storage (Dropbox like), and the possibility to deploy new services such as a set of Linux containers (Docker based).

Routing protocols

One important self-management mechanism in a community network is the routing. We studied, compared and characterized the scalability, performance, and stability of three proactive mesh routing protocols: OLSRv1, BMX6 and Babel, three common routing protocols in wireless community networks, used in the networks involved in our testbed. These routing protocols have been further characterized by studying their control overheads, convergence delay, CPU and memory consumption, and stability. Our experimental evaluation results show the relative merits, costs, and limitations of the three protocols [7]. The results show a very lightweight Babel protocol but too expensive in large, dense and changing networks. In such scenarios OLSRv1 with the MPR mechanism and BMX6 seem to perform better, with lower control overheads for BMX6 but higher memory requirements than other protocols.

Resource Sharing and Isolation

Resource sharing is a critical mechanism for a testbed that allows multiple concurrent experiments, but isolation is also an important issue. Slicing of computing resources is done using Linux containers with mostly limited resources (CPU, storage). An experiment or experimental service takes a slice of the overall resources; a set of slivers (i.e. containers). The network can be virtualized for privacy reasons to isolate experiments from user traffic, or allow experiments to capture a fraction of anonymous production traffic (partially implemented and disabled by default for privacy reasons). Network virtualization can be useful for software-defined networking experiments where a slice contains its own set of virtualized slivers, its OpenFlow controller, and its virtualized network [11].

Self-management

There are various self-managing systems necessary in large and dynamic networks: self-configuration (experiments in adaptive radio channel allocation, power and rate), self-healing (experiments with services that adapt to node or link failures), self-optimization (adaptation according to the available resources depending on internal or external influences). In these areas we have experimented with potential interference detection. Mitigation among different uses can be assessed through a monitoring system that collects detailed metrics about each node, link and sliver, and an experimental resource interference detection and protection system for co-located slivers [13], protecting the most resource sensitive processes in nodes with high utilization.

Business Models, Systems, and Services

Although CNs have already been studied from many angles, there is still insufficient understanding of the practices and methodologies which have given rise to such complex collaborative systems. We have analysed guifi.net in depth [2]. It started in 2004 and is currently the largest CN worldwide with more than 30,000 routers (2015). The guifi.net model is an excellent example to make CNs sustainable and scalable. In fact, guifi.net has received the first ever European Broadband Awards 2015 in the category of innovative model of financing, business and investment.

The fundamental principles of guifi.net, in seeking to be fully inclusive, evolve around i) the openness of access (usage) of the infrastructure, and ii) the openness of participation (construction, operation, governance) in the development of the infrastructure and its community.

These fundamental principles applied to an infrastructure result in a network that is a collective good, socially produced, and governed as a common-pool resource (CPR).

The network is a collective good in which participants contribute their efforts and various goods (routers, links, and servers) which are shared to build a computer network, which combined with several Internet protocols, results in a peer property, provided that the community rules, in the form of a community license, are respected by all participants.

The development of a CN is a social or peer production because the participants work cooperatively, on a local scale, to deploy an infrastructure by

building network islands, and at global scale to share knowledge and coordinate actions to ensure the interoperability of the infrastructure deployed at the local level. The common-pool resource (CPR) is the model chosen to hold and govern the network. The participants must accept the rules to join the network and must contribute to the required infrastructure, whilst always maintaining the ownership of any hardware they contributed and also the right to withdraw.

Nonetheless, as with any other CPR, CNs are fragile. More precisely, they are prone to congestion, as connectivity is subtractable, and subject to the free riding problem, because CNs are intentionally non-excludable. Thus, efficient and effective governance tools are needed in order to protect the core resource from depletion, that is to say, to protect it from the "tragedy of the commons".

To build an effective governance architecture, it is essential to clearly identify the stakeholders based on interests, specific tasks, and potential conflicts of interest. These are volunteers, professionals and their customers, and public administrations. All participants who obtain connectivity must contribute to the infrastructure, directly or indirectly, and can participate in the knowledge creation process.
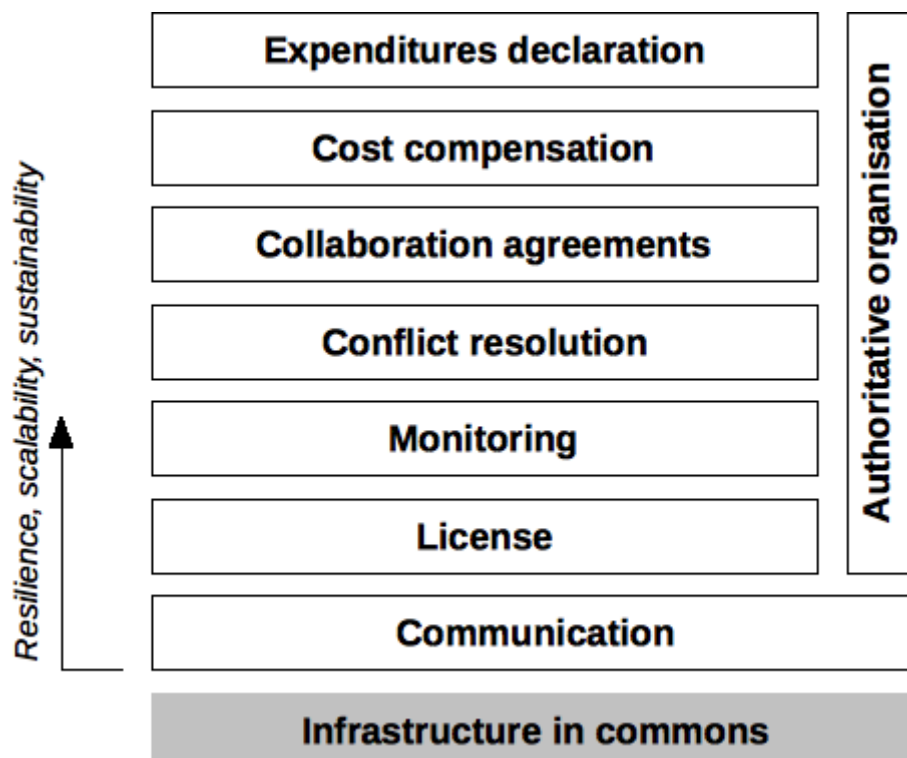

Figure 3: Governance architecture.

Figure 3 depicts the architecture of the governance tools. Effective means of communication are essential for any peer production project. The license establishes the participation framework and sets the boundaries of the CPR. The participation framework must be flexible enough to enable mechanisms for self-sustainability such as economic activity, but at the same time must safeguard the essence of the project. The monitoring system is essential for the network operation and accounting for resource usage. A coherent conflict resolution system avoids arbitrariness and minimizes the number of disputes. Through the mechanism of expenditures declaration, the participants indicate

the resources they contributed which must be accounted for. The collaboration agreement system regulates the for-profit usage of the resources. The economic compensation system balances the contributions that have been accounted for and the resource usage of for-profit participants. Finally, an authoritative organization recognized by all the participants is needed to operate the aforementioned tools and mechanisms.

Opportunities for an Internet for All

Community networks have received growing attention in the last few years as an alternative model used to develop networking infrastructure commons from a local and self-organized perspective, alongside traditional telecom models that cannot serve at least half of the population in the word. We hope to have contributed to this trend. The IRTF GAIA Working Group emerged from our efforts and became an active and global community exploring all models in order to bring connectivity to all. The Internet Society Wireless for Communities program is bringing resources in order to build capacity and help accelerate the development of community networks globally [5]. The Internet Governance Forum is also starting to study community networks for the same purpose. In South Africa, Zenzeleni networks together with UWC and other organizations is creating a second layer organization to expand networking cooperatives across rural South Africa [9]. The ongoing NetCommons.eu project in Europe is looking further at CN from the technical, legal, socio-economic, policy and regulation perspectives for the next three years.

Conclusions

Network infrastructures are needed to provide Internet access to every person in the world, enabling them to participate in our global society and benefit from the many opportunities arising from being online. Underserved communities have found ways to develop their own local network infrastructures. These self-organized infrastructures, developed using commodity network devices and using an open unlicensed spectrum, become a key local infrastructure, managed and governed as a common-pool resource, as with other local community resources for the common interest. These community networks are an emerging and successful model for the Future Internet, with hundreds of examples around the world.

However, these community networks face many challenges. For that reason, a large team of research and community organizations developed Community-Lab.net, which offers unified access to an open testbed with tools that allow researchers to deploy, run, monitor and experiment with services, protocols and applications on real-world community networks.

Research and development in the last five years has produced a large set of results (software, algorithms, models, evaluations) and improvements around resource allocation, routing, cross-layer optimized systems and services, benchmarking methods and tools, and detailed evaluations of social, technological, economic and legal aspects in several developing and developed areas of the world.

Acknowledgement

References

[1] Bart Braem, et al. 2013. "A case for research with and on community networks". *SIGCOMM Comput. Commun. Rev.* 43, 3 (July 2013), 68-73.

[2] Roger Baig, Ramon Roca, Felix Freitag, Leandro Navarro, "guifi.net, a crowdsourced network infrastructure held in common", Computer Networks, Vol 90, 29, 2015.

[3] P. Escrich, Baig, R., Dimogerontakis, E., Carbo, E., Neumann, A., Fonseca, A., Freitag, F., and Navarro, L., "WiBed, a platform for commodity wireless testbeds", in Wireless and Mobile Computing, Networking and Communications (WiMob), 2014 IEEE 10th International Conference on, 2014, pp. 85-91.

[4] Jeroen Avonts et al... (2013). A Questionnaire based Examination of Community Networks. The 9th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2013).

[5] Wireless for Communities: A case book, 2013 Digital Empowerment Foundation.

[6] L. Peterson, T. Anderson, D. Culler, and T. Roscoe. A Blueprint for Introducing Disruptive Technology into the Internet. (HotNets-I '02), October 2002.

[7] Axel Neumann, Ester López, Leandro Navarro, Evaluation of mesh routing protocols for wireless community networks, Computer Networks, Volume 93, Part 2, 24 December 2015, Pages 308-323, ISSN 1389-1286.

[8] Michael Adeyeye, Paul Gardner-Stephen, The Village Telco project: a reliable and practical wireless mesh telephony infrastructure. EURASIP Journal on Wireless Communications and Networking, December 2011, 2011:78

[9] Carlos Rey-Moreno, Zukile Roro, William D. Tucker, Masbulele Jay Siya, Nicola J. Bidwell, and Javier Simo-Reigadas. 2013. Experiences, challenges and lessons from rolling out a rural Wi-Fi mesh network. In Proceedings of the 3rd ACM Symposium on Computing for Development (ACM DEV '13). ACM New York, USA, Article 11, pages 1-10.

[10] A. Neumann, L. Navarro, R. Baig and P. Escrich, "Receiver-driven routing for community mesh networks," World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a, Madrid, 2013, pp. 1-7.

[11] A. Abujoda, D. Dietrich, P. Papadimitriou, A. Sathiaseelan, Software-Defined Wireless Mesh Networks for Internet Access Sharing, Elsevier Computer Networks, October 2015.

[12] Kurtis Heimerl, Shaddi Hasan, Kashif Ali, Eric Brewer, Tapan Parikh. A Longitudinal Study of Local, Sustainable, Small-Scale Cellular Networks. Information Technologies & International Development (ITID) 2015.

[13] Navaneeth Rameshan, Leandro Navarro, Enric Monte, and Vladimir Vlassov. 2014. Stay-Away, protecting sensitive applications from performance interference. In Proceedings of the 15th International Middleware Conference (Middleware '14). ACM, New York, NY, USA, 301-312.

[14] Mennan Selimi, Amin M. Khan, Emmanouil Dimogerontakis, Felix Freitag, and Roger Pueyo-Centelles. 2015. Cloud services in the Guifi.net community network. Computer Networks 93, P2 (December 2015), 373-388.

[15] Christoph Barz, Christoph Fuchs, Jonathan Kirchhoff, Julia Niewiejska, Henning Rogge, OLSRv2 for Community Networks: Using Directional Airtime Metric with external radios, Computer Networks, Volume 93, Part 2, 24 December 2015, Pages 324-341, ISSN 1389-1286, http://dx.doi.org/10.1016/j.comnet.2015.09.022.

Leandro Navarro is associate Professor at the Computer Architecture Department of UPC. He received his Ph.D. from UPC in 1992. Research interests include the design of scalable and cooperative Internet services and applications. He coordinates the CNDS (Computer Networks and Distributed Systems) research group at UPC. He coordinated the CONFINE IP project and currently coordinates the Erasmus Mundus Joint Doctorate in Distributed Computing (EMJD-DC). He is council member of the Association for Progressive Communications (apc.org).

Dr. Bart Braem holds a PhD in Computer Science from the university of Antwerp, He is currently Technology Lead of the iMinds smart cities project City of Things where he studies large-scale experimentation on IoT in realistic conditions. His research interests include smart cities, large-scale networking and IoT. Bart was the research WP leader of the EU FIRE project CONFINE, which developed and operated a community networks testbed.

Christoph Barz is a senior researcher at the Fraunhofer FKIE in the field of communication systems. He is working on secure and robust IP-based communications solutions and QoS for heterogeneous wireless networks as well as on their integration into core network infrastructures. Mr. Barz received his degree in computer science with a focus on communications systems at the University of Bonn in 2002. Until 2009, he was a research assistant at the University of Bonn.

Felix Freitag is associate Professor at the Computer Architecture Department which he joined in 1999. PhD in Telecommunication Engineering from the UPC

in 1998 in the area of speech recognition, neural networks, statistical modelling. His research interests include the performance evaluation of distributed systems, integration of virtualization, and mechanisms for resource allocation in distributed systems. He was leading the EU funded project Clommunity on Community Clouds.

Roger Baig Viñas joined in 2011 the Foundation for the Neutral, Free and Open Network - guifi.net as a researcher and currently is actively involved in two EU projects, RIFE and NetCommons. He has been volunteering in guifi.net community network in 2006. There he has actively contributed in many aspects such as event, projects and talks. His research interests are on mesh routing protocols, socio-economics and management of community networks.

Joseph Bonicioli has walked the IT arena for more than fifteen years. Since 2011 serves for the second time as the President of the Non-Profit Association Athens Wireless Metropolitan Network. Since October 2011 has been responsible for AWMN participation in the EU FP7 funded research project (IP) CONFINE. With a both technical and business minded attitude, he has been involved in projects involving wireless technology, Community Networks, Cloud Computing, Network Security, High Availability among others.

Ivan Vilata-i-Balaguer is a computing engineer with a focus on networking, virtualization and distributed systems based on Free Software and open standards and technologies, as well as the sociological impact of these in empowering people. He has experience in developing, designing and administering systems for ISPs, NGOs and SMEs. He has taken a main role in the design of the CONFINE architecture.
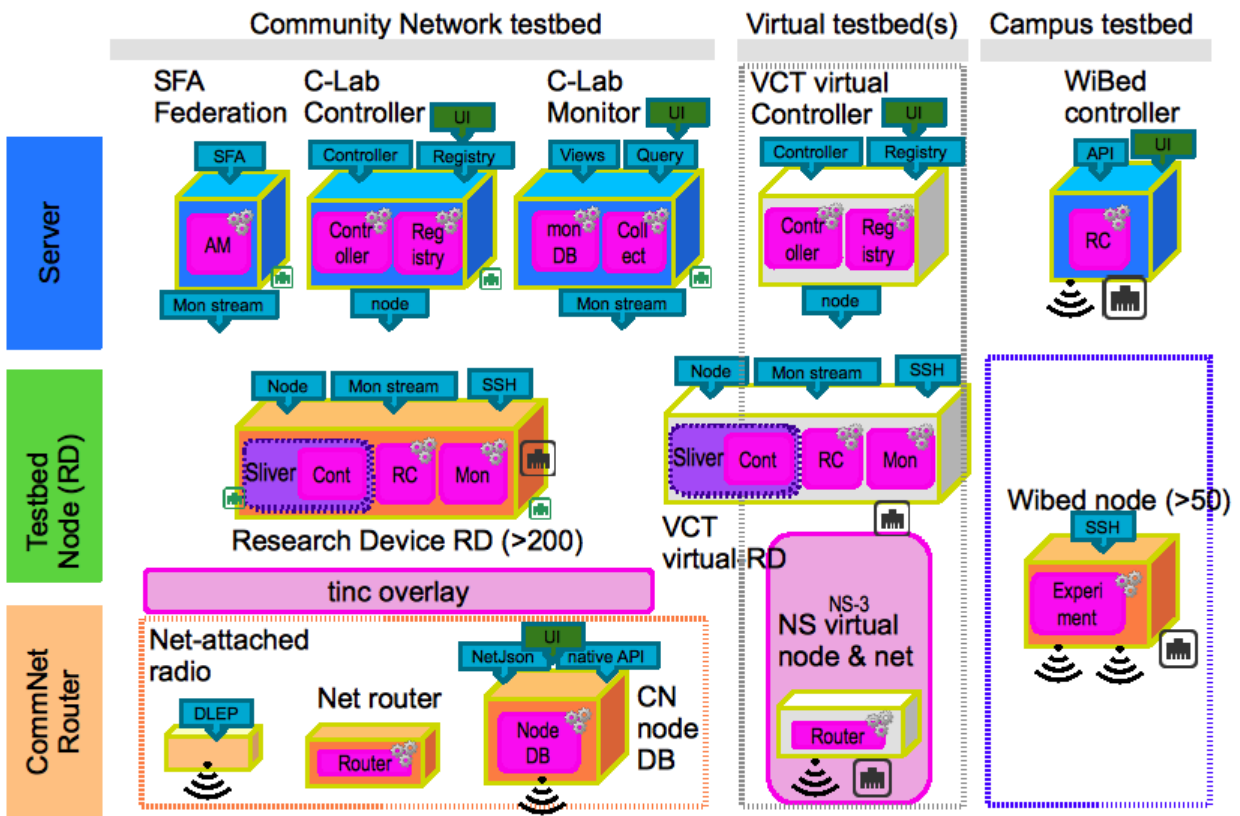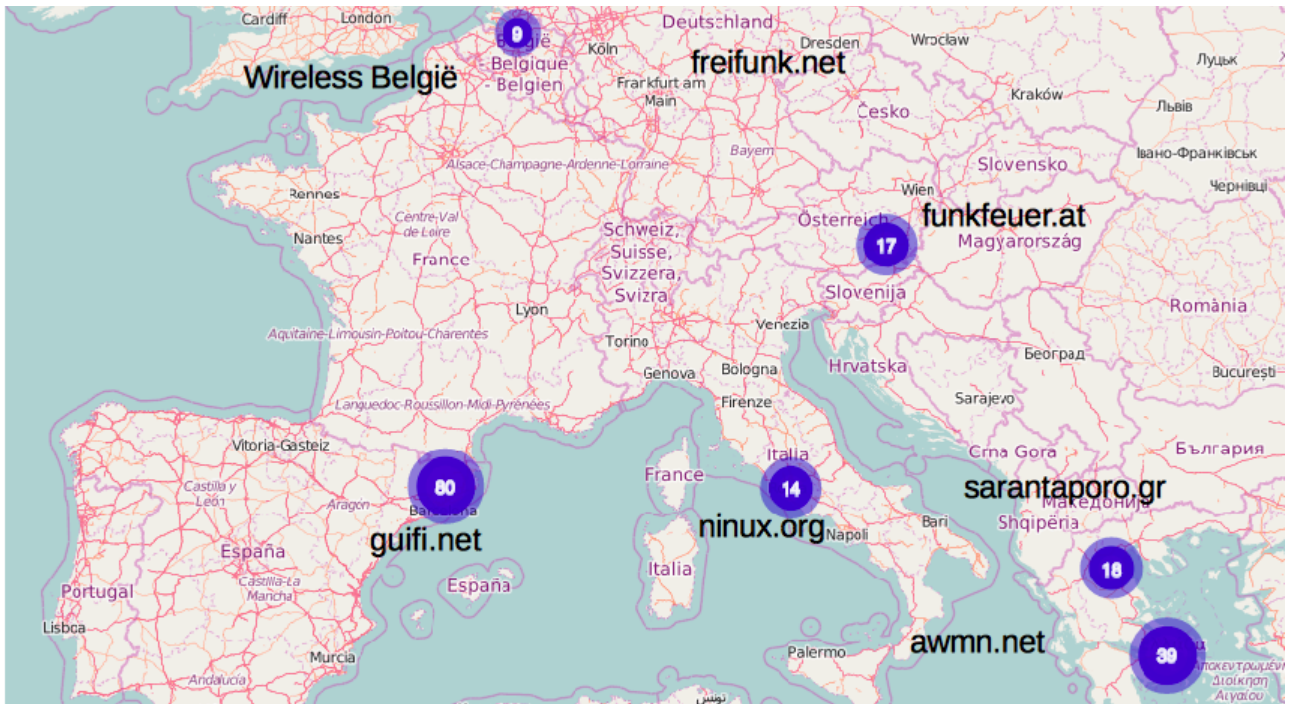
Figure 1: The node/functional architecture of the Community-Lab (C-Lab) testbed
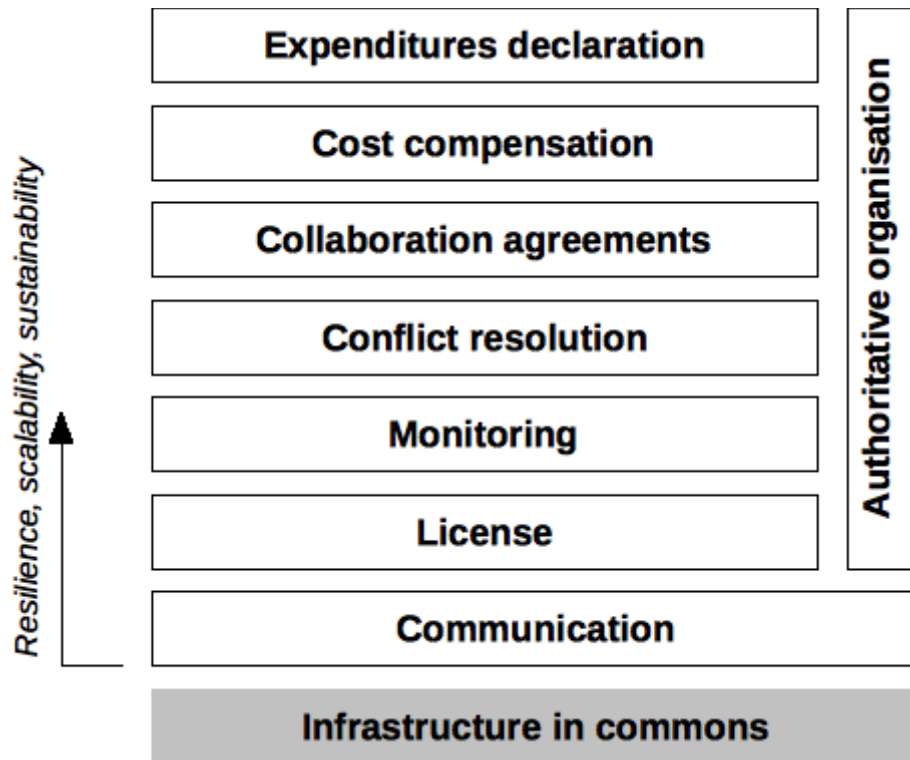
Figure 2. The Community-Lab Community Network testbed (2015).

Figure 3: Governance architecture.