404 Not Found? – A Quest For DTN Applications

Jörg Ott
Aalto University
Department of Communications and Networking
Otakaari 5A, 02150 Espoo, Finland
jo@comnet.tkk.fi

Categories and Subject Descriptors

C.2.2 [**Computer Communication Networks**]: Network Protocols; C.2.4 [**Distributed Systems**]: Distributed Applications

Keywords

Delay-tolerant Networking; DTN; Routing; Mobility, Applications

1. EXTENDED ABSTRACT

Delay-tolerant Networking (DTN) [3] has moved a long way from its—for many probably somewhat elite—origin of a technology for an Interplanetary Internet in the late 1990s to an established research discipline. One major contributor to this trend was the observation that quite a few terrestrial networks exhibit delay-tolerant properties, albeit of different nature: from sparse mobile ad-hoc to sensor networks to mobile Internet access, we find delay tolerance as an important element to describe communication behavior and to design protocols suitable for operation in the respective challenged networking environment. And even fixed infrastructure networks may benefit from delay-tolerant approaches to data transmission [7]—as did UUNet decades ago, when forwarding mails and news in a multi-hop store-and-forward fashion enabled communication involving machines that were not "always on" in the first place.

Delay-tolerant networking has contributed to our understanding of networking at least in a twofold manner:

- It has helped extending the reach of communication into areas previously beyond the grasp of generic networking architectures (as opposed to closed application-specific solutions that may have existed in some areas) [3].
- We have learned reconsidering protocol design practices to be able to build systems for DTNs,[1] which are also applicable to the Internet at large: from robustness in the presence of disruptions to separating functions for application data units from their delivery protocols (e.g., securing the objects, not the transport) to storing (i.e., caching) meaningful (and identifiable) information units [11].

The former had a substantial impact on mobile ad-hoc networking research for which mostly connected networks (with high node density) had to be assumed, a requirement that could be relaxed with delay tolerance so that MANETs got much closer to reality.

[1]Various issues have arisen independently in other contexts.

MobiOpp'12, March 15–16, 2012, Zürich, Switzerland.
ACM 978-1-4503-1208-0/12/03.

While also applicable to sparse vehicular networks or wireless sensor networks, enabling networking between mobile devices, also dubbed *opportunistic* or *pocket-switched networking* [5], has been the primary driver for research in a number of fields. These include studying contact patterns between mobile devices carried by humans, understanding human mobility (as far as relevant for device-to-device communication) and developing models that can reproduce the observed contact patterns, and correlating the contacts and social networks to allow prediction of future interactions.

These and others feed into system and protocol design for mostly infrastructure-less communication between mobile nodes and their respective evaluation. Dozens of routing protocols were developed and diverse communication paradigms beyond unicast adapted to mobile DTNs, including multicast, publish/subscribe, and broadcast, with many different flavors being developed. Recently, plain opportunistic communication got another layer on top: service discovery and having tasks executed by individual or groups of other nodes to later collect the results: *opportunistic computing* [2] or *crowd computing* [10], in analogy to cloud computing.

So, here we are with all these models and protocols and systems. But what to do with them? Once in a while it feels that, with delay-tolerant networking, we are heading for the same trap that the MANET community has fallen into before: designing (routing) protocols without any clear applications in mind that might use them or defining more or less artificial problems with little bearing on reality. Disaster management, crisis scenarios, and communication in remote areas are prime examples for application scenarios to be found in grant applications. Yes, these scenarios do matter, but they cannot be all there is. We should do better—and we can!

However, we must always remember what or whom we are designing for and which requirements arise from this target. For example, a lot of work has explored human-to-human communication or Internet access in opportunistic environments: for web page retrieval, web and vicinity search, twitter, and messaging, among others. But very little has been done to understand how such applications would need to be designed to become actually usable and useful. We often tend to be happy about (or blinded by?) performance figures that show that something works *in principle* without questioning too much the implications *in practice*. The latter particularly includes appreciating that the delay tolerance of the user and her interaction with the system (and thus the user interface).

This touches a key point concerning opportunistic networks: the lack of predictability. If users want something, they want it now! At the very least, they want to know when to expect a response. "404 Not Found" or a browser's note "Server not found" are clear, whereas an hourglass or a similar indication to wait leaves the user in the dark about her chances for success. Opportunistic networking environments in which message delivery latency can be anything between one second and several days—or infinity since odds are that a message is not delivered at all—deny a user this very predictability.[2] Users calibrate their expectations anyway when directly connected to the Internet, when search results or the initial bits of web page contents are expected to appear within a second—and our work culture has evolved to expect even asynchronous means for information exchange such as email to work more or less instantaneously.

All this makes building compelling DTN applications a challenging task, given such demanding competition especially in places with pretty good wireless or cellular coverage and flat rates for mobile data at least for local subscribers. We have basically two options: 1) Reconsider the way we think about application interaction and communication paradigms that can satisfy mainstream user expectations. 2) Find those *niches* where the mainstream does not matter. The aforementioned disaster, crisis, and rural communication applications fall into this latter category.

For the mainstream, we face the challenge of designing applications that either cannot be built using omnipresent wireless infrastructure and backend services in the cloud or try to minimize leaking information to such services. Exchanging large volumes of information at high data rates or in privacy, implicitly provided by physical proximity to some extent, could thus be important drivers.

One obvious use is **content sharing**. DTN content distribution has been pioneered by the *PodNet* system [8] that offers users to share content in a peer-to-peer fashion according to their interests. *Floating Content* [12] restricts content dissemination to a predefined geographic area for each content item and thus exploits locality, supports spatial and temporal re-use of node buffers, and implicitly limits resource consumption. *SCAMPImusic* allows experiencing the music tastes of a user's immediate surroundings by sharing music contents in a volatile way.[3] From a usability perspective it is important that these applications all work in the background, collecting content according to a user's preferences: when they catch the user's attention, they only present what they already got, thus preserving the instant interaction a user would expect. Moreover, as there is no way for a user to know which content to expect in each case, no expectations (e.g., completeness of the available music) can be disappointed.

Content sharing between mobiles may **complement infrastructure** usage. Devices may perform opportunistic caching and ask their neighbors about locally available content before downloading via the infrastructure. This may *offload* data from the infrastructure, but also assist in preserving privacy, e.g., when sharing map tiles of the environment so that server cannot track users [1]. Such applications would operate in the background invisible to the user.

Niche applications have contributed substantially to DTN development, especially before mobile phones were close enough to the capabilities required for ad-hoc interactions between them. In the past, quite some focus was on **sensing**, e.g., for environmental conditions [9] or to study animal behavior [6], where sensor data were replicated between nodes and collected at dedicated places, using animal, vehicle, or human mobility for data carriage. Many further applications to extend the reach of networking for different areas (underwater, mountains, aerial surveillance) continue to emerge.

Another niche application area has received rather limited (academic) attention so far: various **industrial environments** may offer quite similar communication challenges as mountainous or rural areas, such as lack of (deployable) infrastructure, sparse node density, and too limited communication range. They would benefit from delay-tolerant networking to improve (if not enable) communications. One example is networking people, equipment, and a control room in underground mines [4], where rock blocks wireless communication effectively. Mines have usually very limited facilities for data communications and, if available, cover only a fraction of the total mine. Yet orders need to be sent to and status information collected from potentially hundreds of machines so that progress monitoring and planning can proceed efficiently. Leveraging vehicles and workers equipped with mobile devices as message carriers can improve information availability substantially. Such constrained environments, where DTN can make a difference in practice, can likely be found elsewhere.

While niche applications may benefit from dedicated hardware and closed deployments, which simplifies configuration and operations, mainstream applications face an open and heterogeneous environment. Especially with content sharing, making participation legally safe, technically robust (e.g., against malware), and privacy preserving are challenges yet to be addressed.

One common observation across all applications is that they are usually applied in a specific context and those observations and requirements don't find their way into other researchers' evaluations too often. So, we need to break out of these research silos to share (and use!) insights about meaningful scenarios, applications, and parameters more broadly: too often, we find evaluations for routing protocols that use traffic with little relation to applications or a rather limited set of mobility models. And evaluations are often quite optimistic about lower layer characteristics from detection times, to pairing success rates, to transmission rate.

Finally, it is difficult to obtain real usage patterns for DTN applications without developing, deploying and measuring them. We need to "eat (more of) our own dog food" to gain a better understanding of how those applications and their underpinnings work for real. If we don't, who would?

[2]This also applies to the aforementioned applications: e.g., people in emergency situations want to know if somebody heard them and need a "voice" to tell them that help is on the way.

[3]Idea and prototype developed by Teemu Kärkkäinen in the EC FP7 project SCAMPI (grant agreement 258414).

2. REFERENCES

[1] AMINI, S., LINDQVIST, J., HONG, J., LIN, J., TOCH, E., AND SADEH, N. Caché: Caching location-enhanced content to improve user privacy. In *Proc. of ACM MobiSys* (2011).

[2] CONTI, M., GIORDANO, S., MAY, M., AND PASSARELLA, A. From Opportunistic Communication to Opportunistic Computing. *IEEE Communications Magazine 48*, 9 (September 2010), 126–139.

[3] FALL, K. A Delay-Tolerant Network Architecture for Challenged Internets. In *Proceedings of ACM SIGCOMM* (August 2003).

[4] GINZBOORG, P., KÄRKKÄINEN, T., RUOTSALAINEN, A., ANDERSSON, M., AND OTT, J. DTN communication in a Mine. In *Proc. of ExtremeCom* (September 2010).

[5] HUI, P., CHAINTREAU, A., SCOTT, J., GASS, R., CROWCROFT, J., AND DIOT, C. Pocket Switched Networks and Human Mobility in Conference Environments. In *ACM WDTN Workshop* (2005).

[6] JUANG, P., OKI, H., WANG, Y., MARTONOSI, M., PEH, L. S., AND RUBENSTEIN, D. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. In *Proc. of ASPLOS-X* (October 2002).

[7] LAOUTARIS, N., AND RODRIGUEZ, P. Good Things Come to Those Who (Can) Wait. In *Proc. of ACM HotNets VII* (2008).

[8] LENDERS, V., MAY, M., KARLSSON, G., AND WACHA, C. Wireless ad hoc podcasting. *ACM/SIGMOBILE Mobile Comp. and Comm. Rev. 12*, 1 (2008).

[9] MCDONALD, P., GERAGHTY, D., HUMPHREYS, I., FARRELL, S., AND CAHILL, V. Sensor Network with Delay Tolerance (SeNDT). In *Proc. of IEEE ICCCN* (2007).

[10] MURRAY, D. E., YONEKI, E., CROWCROFT, J., AND HAND, S. The Case for Crowd Computing. In *Proc. of ACM MobiHeld* (2010).

[11] OTT, J. Delay Tolerance and the Future Internet. In *Proc. of IEEE WPMC* (2008).

[12] OTT, J., HYYTIÄ, E., LASSILA, P., VAEGS, T., AND KANGASHARJU, J. Floating Content: Information Sharing in Urban Areas. In *Proc. of IEEE Percom* (2011).