Delay-Tolerant Management Using Self-* Properties and P2P Technology

Jéferson Campos Nobre, Pedro Arthur Pinheiro Rosa Duarte,
Lisandro Zambenedetti Granville, Liane Margarida Rockenbach Tarouco
Institute of Informatics – Federal University of Rio Grande do Sul
Porto Alegre, Brazil
Email: {jcnobre, paprduarte, granville, liane}@inf.ufrgs.br

***Abstract*—The introduction of self-* properties has been proven to be a feasible approach for the management demands of Delay-Tolerant Networks (DTNs). Among the properties of the self-* management vision, self-healing figures as a key property in improving the dependability of the managed infrastructures. An interesting possibility to materialize self-* support in delay-tolerant management is through the employment of Peer-to-Peer (P2P) technology. In this paper, we introduce a P2P self-healing service tailored for delay-tolerant management. We implemented the proposed service using ManP2P-ng, an open source P2P-based network management system.**

I. Introduction

Computer networks have evolved in size, complexity, and heterogeneity in the last years. Besides, the requirements of the services provided in these networks have become increasingly diverse. In this scenario, solutions to manage the underlying communication infrastructure and help network human administrators in their daily tasks are crucial. However, these solutions rely on some premises, such as low-latency data exchange and permanent connectivity, that sometimes can not be met. Some example of networks without these premises are Internet access for remote villages, Vehicular networks, "Educational Projects" networks, and InterPlanetary Internet.

The introduction of a Delay-Tolerant Network (DTN) architecture can provide support to an wide variety of underlying communication technologies, multiple and simultaneous routing paths, hop-by-hop and end-to-end reliability, and disconnection/disruption tolerance [1]. However, little work has yet been done on DTN Management (DTNM), despite it seems essential in several scenarios [2]. Some works address restricted scenarios [3], but more general solutions have not arrived.

The introduction of distributed technologies in network management has leaded to improvements over the traditional centralized approach, for example, when communication networks grown in size. Indeed, some authors have already proposed the utilization of traditional Distributed Network Management (DNM) approaches to manage DTNs [3]. However, the complexity of DTNs demands management features that are not present, at least not explicitly, in usual DNM technologies (*e.g.*, support for intermittent connectivity). An alternative that complements traditional DNM technologies consists in the employment of Peer-to-Peer (P2P) management overlays. Such overlays merge characteristics of P2P and network management systems, better enabling, for example, a more robust connectivity among management entities [4].

P2P-based Network Management (P2PBNM) has interesting properties for the management of DTNs, such as the utilization of local data and autonomy of management entities. In fact, several DTNs applications are intrinsic P2P. These two technologies also have analogous characteristics, such as robustness features. Furthermore, P2PBNM also enables the embedding of self-* properties into network management. In this context, some authors claim that self-* properties are essential to DTNM [3]. Self-healing is one of the key self-* properties, which aims at automating the failure detection and handling, improving the dependability of the communication infrastructures. In the context of DTNM, self-healing features have one more appeal: healing procedures can be performed without having to communicate with a centralized entity, such as a management station. This communication can be intermittent and/or significant slow on DTNs.

In this paper, we propose a self-healing service for DTNs using P2P technologies. We present an extension of a prototype P2PBNM system to support overlay operations in face of delays and disruptions. Our self-healing service exploits the support for delay-tolerant transport and grouping of management messages. This service can be contracted at any time by system administrators to monitor and heal managed elements of a DTN. The main contribution of our work is to provide self-healing features to manage elements of DTNs regardless of the existence of centralized managers. We also show that it is possible to reuse management applications present on P2PBNM systems in DTN environments.

The remaining of this paper is organized as follows. In Section II, we present the main concepts related to DTNs and self-* properties in P2PBNM approaches. In Section III, our proposal and its associated concepts are described. The details of a proof-of-concept implementation are presented in Section IV. Finally, conclusions and future work are provided in Section V.

II. Background

In this section, we first discuss the fundamentals regarding Delay-Tolerant Networks (DTNs) as well as the concepts behind HTTP-DTN. Afterwards, we review the state-of-the-art on self-* properties in P2P-Based Network Management (P2PBNM). These properties improve the desirable characteristics of P2PBNM features in delay-tolerant management.

978-3-901882-50-0

A. Delay-tolerant networks and HTTP-DTN

Nowadays, new classes of non-conventional computer networks have arisen in consequence of technological research and development. In these networks, one can observe that some assumptions of the TCP/IP architecture are relaxed, such as the existence of an end-to-end path between hosts, a non-excessive maximum round trip time, and a small probability of having packets discarded. Networks in which these assumptions cannot be met are embraced in the *Delay Tolerant Network* (DTN) concept [1].

Several "clean slate" protocols for DTN delivery have been already proposed (*e.g.*, Bundle Protocol [5]). However, some authors suggest the reuse of existing standards in order to simplify the deployment of DTNs [2]. One of these suggestions consists in exploiting the HyperText Transfer Protocol (HTTP) in DTN contexts as a transport layer independent, session protocol among communicating DTN nodes (hop-by-hop). This HTTP usage is referred to as HTTP-DTN [2].

With HTTP-DTN, applications running on each node in the network communicate with their peers using dedicated, store-and-forward, hop-by-hop HTTP transfers to effectively deliver data [2]. HTTP is simply adapted to DTN environments by the use of HTTP/1.1 with persistence, pipelining, and some extra headers. These headers define an independent addressing schema, thus alternatives technologies (*e.g.*, Saratoga [6]) are enabled as a transport for HTTP messages.

Messages can be grouped in HTTP-DTN through the use of *Package-** fields; these groups are thus called *packages* (in an analogous way to the use of *bundles* in the Bundle Protocol [5]). Grouping messages is useful in DTNs because it can save scarce resources such as bandwidth. For example, multiple original message headers can be reduced to the group's single one [2].

B. Self-* properties in P2P-Based Network Management

The use of P2P technology in network management, also known as P2P-Based Network Management (P2PBNM), is an approach to integrate characteristics of Distributed Network Management (DNM) and P2P overlays [4]. Many features required in distributed management are intrinsically provided by P2P overlays, since these overlays are developed to efficiently use different resources that are distributed in several peers. In addition, P2PBNM systems use Application Layer Routing (ALR), which adapts more easily to unconventional environments such as DTNs.

In P2PBNM, management tasks are performed by peers of the P2P management overlay through management components. These components advertise the tasks they perform as management services [7]. Moreover, peers are organized into groups according to their management services [7], *i.e.*, peers that offer a specific management service are part of the same peer group.

P2PBNM eases the introduction of self-* properties in network management. These properties can be effectively deployed using concepts from P2PBNM, such as management components and services. In this context, some initiatives investigated the joint use of self-* properties in P2PBNM. In a previous work of our research group, Duarte *et al.* [8] proposed a self-healing mechanism used to manage both single elements and whole infrastructures through the use of monitoring and healing workplans. DTNs were not supported, however, because the self-healing mechanism operated assuming that network connections were stable, which is not the case in DTNs.

In summary, the joint use of P2P overlays and self-* properties in network management offers improvements in different aspects, such as scalability and reliability. Despite these improvements, P2PBNM cannot be directly used in DTN management without proper adaptations. As mentioned before, for example, current P2P self-healing mechanisms break in the presence of intermittent connectivity because they sustain control loops among management nodes over supposedly stable connections.

III. A P2P-Based Network Management extension for delay-tolerant self-healing

The application of P2P technologies improves the dependability of network management system (*e.g.*, through the replication of management components). P2P-based Network Management (P2PBNM) systems extend distributed network management systems through the composition of characteristics of distributed management models and P2P overlays. However, in networks where the link delay/disruption is dominant, these technologies need adaptations to maintain this dependability. Besides that, in Delay-Tolerant Networks (DTNs), network management solutions cannot rely on low-latency data exchange, thus it is necessary alternative mechanisms to maintain control of network resources. These solutions also have to cope with some self-management since the propagation time necessary to send operational commands can prevent some network managers abilities of performing remote procedures, such as remote healing.

We propose to support delay-tolerant management by introducing an extension on P2PBNM systems. Besides that, we explain the concepts behind a delay-tolerant self-healing service tailored for this extension. We first describe this P2PBNM extension. Afterwards, the delay-tolerant self-healing service is presented.

A. A P2PBNM extension for delay-tolerant management

The P2PBNM extension promotes two processes in each peer that forms the management overlay: the maintenance of the management overlay, now operating over a delay-tolerant environment; and the intermediation on the execution of delay-tolerant management tasks. This extension could be used to transport commands (*e.g.*, configuration settings) and reports (*e.g.*, performance data) in order to perform simple delay-tolerant management tasks; it can also enable the realization of more sophisticated management tasks, such as the delay-tolerant self-healing of failed managed entities.

The P2PBNM extension deals with network delays and disruptions by using store-and-forward techniques and adapting delay-tolerant protocols to overlay operation. The extension is not tied to any delay-tolerant or IP-related protocols, thus it is possible to wrap different delay-tolerant protocols (*e.g.*, bundle protocol) through its facilities. The design choice to keep the extension not tied to specific protocols also supports transport layer independence, which makes it suitable for networks with different connectivity properties.

There are some proposed protocols for delay-tolerant delivery. A delay-tolerant protocol that can be employed by the P2PBNM extension is the HTTP-DTN [9]. As described in Section II, the HTTP-DTN is used to enable indirect delay-tolerant communications between source and destination hosts. These communications are performed using intermediate hop-by-hop HTTP-DTN sessions that are established among intermediate hosts. HTTP is ubiquitous and well-understood, what helps its utilization on delay-tolerant scenarios.

The P2PBNM extension enables delay-tolerant management tasks to be performed through different strategies. These strategies define how management data is produced, stored, and transmitted. Among these strategies, *pull* (management requests by clients and replies by servers at managed devices) and *push* (server pushes management data to clients) strategies can be used [10]. When the HTTP-DTN is used by the P2PBNM extension, pull and push interactions are accomplished through the use of GET and PUT requests, respectively. Pushed objects are cached and then relayed by intermediate peers through PUT requests until they reach their destinations. Analogously, pulled objects can be forwarded by intermediate peers to their destinations through GET requests [9].

The P2PBNM extension for delay-tolerant management also enables message grouping. We propose using HTTP-DTN and its native support for grouping messages in packages (as described in Section II) to improve P2P communications of the management overlay, specially in environments with restricted connectivity, such as DTNs. The policies that define how to group management messages in HTTP-DTN packages are enforced by our proposed extension and specified by human administrators.

B. A delay-tolerant self-healing service

The P2PBNM extension for delay-tolerant management proposed in the present Section provides a message delivery service transparent of delays and disruptions for management peers. However, beyond the features offered by the proposed extension, a self-healing service still has to be tailored to deal with DTN environments. From now on, we present the adaptations over the self-healing service, which enable the monitoring and healing of managed entities in DTNs.

In a previous work [8] we have introduced a P2P self-healing service for conventional networks based on *monitoring and healing workplans*. Such workplans compile, using a well-defined language, the administrator's knowledge on how to monitor and heal the managed infrastructure. Although workplans do not require a specific network technology, our original self-healing mechanism cannot be directly deployed in a delay-tolerant environment because it assumes stable and low-delay network connections. When required, the human administrator calls the self-healing service informing monitoring and healing workplans, as well as the target managed entities to be observed and eventually healed.

Monitoring workplan execution occurs concurrently, without further synchronization or coordination among the peers which host it since the instability of end-to-end paths would either prevent or largely retard its scheduling. The result of the execution of a monitoring workplan is the diagnosis of the operational status of managed entities. If this diagnosis indicates that a target managed entity is healthy, the workplan is rescheduled for another round of execution. However, if the diagnosis indicates that the managed entity is unhealthy, the monitoring peers that noticed the fault send an unhealthy notification to healer peers.

A healing workplan starts to be executed by the healing peers as soon as a number of unhealthy notifications equal to a unhealthy threshold is reached. Once the healing workplan is finished and reaches a positive result, healing peers notify back the monitoring peers. This notification enables the monitoring peers to resume their workplans so the target entities start being observed again. If the healing workplan, however, cannot heal the target entities, the healing peers request the monitoring ones to completely stop the previously paused monitoring workplan and to notify the human administrator about the unsuccessful attempt to heal the target entities. The human administrator has then the opportunity to manually react to that.

IV. Implementation

We implemented the P2PBNM extension and the delay-tolerant self-healing service previously described in Section III as a proof of concept. The implementation was built on top of ManP2P-ng[1], which is an open source P2PBNM system that follows the service-oriented approach. ManP2P-ng has embedded mechanisms for the self-organization and maintenance of P2P overlay network, the same way as a resource discovery protocol. Beyond these, ManP2P-ng provides a Application Programming Interface (API) that may be use to extend its management overlay.

The P2PBNM extension is implemented as a delay-tolerant transport adapter in networking interfaces of ManP2P-ng. This is done to enable the operation of the P2P management overlay in DTN environments. We initially chose to implement the adapter using HTTP-DTN [2] because we were primary interested in opportunistic connections (*i.e.*, networks with frequently disruptions) without long delays, and HTTP-DTN is tailored to deal with this kind of connection. Besides that, the adapter uses MIME to describe messages exchanged among DTN peers.

Addressing aspects of the delay-tolerant transport adapter are not tied to IP or IP-related protocols (*e.g.*, TCP or DNS).Thus, the adapter is able to reuse ManP2P-ng own stack and routing namespace. In this context, the ManP2P-ng address structure is used to bind against HTTP-DTN addressing. Thus, peer addressing information in the current implementation is provided through HTTP-DTN specific header fields. These fields operate using HTTP/1.1-compliant specifications. For addressing purposes, *Content-Destination* and *Content-Source* fields identify the destination and the source of the data, which are filled with peers identifiers. Thus, it is possible to rely on ManP2P-ng routing and address capabilities for message delivering even in a DTN scenario.

[1] ManP2P-ng - http://projects-redes.inf.ufrgs.br/gf/project/manp2png/

Management tasks are performed in ManP2P-ng through management components which implement management services. Management components, and their associated management services, may vary from very simple ones (*e.g.*, a protocol gateway to access management devices via SSH) to more complex ones (*e.g.*, support for autonomic management tasks). These services have a unique service identifier (also known as group identifier). The use of ManP2P-ng API allows instantiation, initialization, and operation of management components. Thus, a management component is managed through a well defined life cycle. This API also supports the use of P2P services provided by the underlying P2P system or by other management components. Moreover, a management component may extend the API, providing extra features to other components.

Different overlay maintenance messages or management messages (due to management tasks) from the same source to the same destination can be grouped into a "bundle" (in the same sense that is done in the Bundle Protocol). The ability to group objects together can be useful in DTN scenarios. This grouping procedure is performed using *Package-** fields, proposed in HTTP-DTN, thus these groups are called *packages*. The criteria to group messages in a package are defined by management components (for management messages) or through the configuration of P2P overlay services (for overlay maintenance messages). Initially, the policy to control which management messages may be grouped in HTTP-DTN packages is defined according to the source and destination fields.

The self-healing mechanism is implemented in ManP2P-ng through two management components: the monitoring service and the healing service components. Despite the fact that overlay characteristics are abstracted by ManP2P-ng P2P services and the P2PBNM extension, both components require some adaptations to properly perform in the presence of delays and disruptions. The adaptations implemented for the self-healing mechanism mainly reflect the connectivity characteristics of DTN environments. In order to deal with locality matters, it is possible to choose the nearest monitoring and healing peer groups using an optional given neighbor. As such, it is now possible to encompass many management entities, as well as their specific data, in one service request by providing a list of managed entities. It is also possible to tune scheduling and threshold parameters considering specific DTNs.

One of the key features of ManP2P-ng architecture is the notion of peer group, where a set of management components implementing the same management service are grouped together. The multiple instances of the same management component at different peers of a peer group provide fault tolerance features inherent to P2P systems. Currently, a peer group with DTN support requires that all participating peers (of the peer group) have this support in our implementation. In other words, the DTN extension works according with the premise that all peers in the peer group are in a "DTN island". However, it is feasible to develop a DTN gateway-like feature (in order to interconnect networks with different characteristics) through adaptations in the DTN extension.

V. Conclusions and Future Work

Delay-Tolerant Networks (DTNs) technology imposes new and novel challenges for network management. Usual techniques employed by management solutions, *e.g.* control loops, cannot be directly applied on DTN scenarios since long delays in data delivery and frequent connection disruptions explicitly defy their basic premises. Furthermore, the self-* properties and self-management are implicit requirements of these networks since the same challenges which prevents the straightaway employment of current solutions also prevents administrators to promptly manage these networks.

In this paper we have initially argued that the intrinsic characteristics of P2P-Based Network Management (P2PBNM) makes it a suitable candidate for a common substrate to manage DTN environments. We then introduced an extension for P2PBNM systems which uses HTTP-DTN in order to provide store-and-forward capabilities as well as message packaging. Besides the aforementioned, this extension permits the reuse of management components already present in P2PBNM systems. In this context, we detailed the adaptation of a self-healing service using the prototype P2PBNM system ManP2P-ng.

The research for DTN management techniques is in its early days and there is plenty of room for improving the work described in this paper. For example, the security features of our proposed solution also need improvement, since in many DTN scenarios these features are mandatory, such as in military DTN environments. Thus, we intend to investigate the use of HTTPS and S/MIME for improving the hop-by-hop and end-to-end security in the management communication sessions.

References

[1] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-tolerant networking architecture," RFC 4838 (Informational), Internet Engineering Task Force, April 2007.

[2] L. Wood, W. Eddy, and P. Holliday, "A bundle of problems," in *Proceedings of the IEEE Aerospace conference 2009*, 2009, pp. 1–17.

[3] C. Peoples, G. Parr, B. Scotney, and A. Moore, "Context-aware policy-based framework for self-management in delay-tolerant networks: A case study for deep space exploration," *Communications Magazine, IEEE*, vol. 48, no. 7, pp. 102–109, 2010.

[4] L. Z. Granville, D. M. da Rosa, A. Panisson, C. Melchiors, M. J. B. Almeida, and L. M. R. Tarouco, "Managing computer networks using peer-to-peer technologies," *Communications Magazine, IEEE*, vol. 43, no. 10, pp. 62–68, 2005.

[5] K. Scott and S. Burleigh, "Bundle protocol specification," RFC 5050 (Experimental), Internet Engineering Task Force, 2007.

[6] L. Wood, W. Eddy, C. Smith, W. Ivancic, and C. Jackson, "Saratoga: A Scalable File Transfer Protocol," *work in progress as an internet-draft, draft-wood-tsvwg-saratoga-11*, 2012.

[7] A. Panisson, D. M. da Rosa, C. Melchiors, L. Z. Granville, Maria, and Liane, "Designing the Architecture of P2P-Based Network Management Systems," in *Proceedings of the 11th IEEE Symposium on Computers and Communications*. IEEE Computer Society, 2006, pp. 69–75.

[8] P. A. P. R. Duarte, J. C. Nobre, L. Z. Granville, and L. M. R. Tarouco, "A P2P-Based Self-Healing Service for Network Maintenance," in *Proceedings of the 12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011), Dublin, Ireland. May 2011.*, 2011.

[9] L. Wood and P. Holliday, "Using HTTP for delivery in Delay/Disruption-Tolerant Networks," *work in progress as an internet-draft, draft-wood-dtnrg-http-dtn-delivery*, November 2010.

[10] E. Birrane and R. G. Cole, "Management of Disruption-Tolerant Networks: A Systems Engineering Approach," in *Proceedings of the SpaceOps 2010 Conference, Alabama, United States. April 2010*, 2010.