Flow Context Tags: Concepts and Applications

Roel Ocampo[1,2], Alex Galis[2], Hermann De Meer[3], and Chris Todd[2]

[1] Department of Electrical and Electronics Engineering, University of the Philippines, Diliman, Quezon City, 1101 Philippines
[2] Department of Electronic and Electrical Engineering, University College London, Torrington Place, London WC1E 7JE, U.K.
[3] Faculty of Mathematics and Computer Science, University of Passau, 94032 Passau, Germany

Abstract. Context awareness can help build dynamic networks by enabling them to automatically adapt to the user's activities, computational environment, and network conditions. Our approach in building context-aware networks uses flow context: information about the intrinsic and low-level characteristics of flows, as well as the nature of the applications, devices, and the activities, intentions, preferences or identities of the users that produce or consume them. We tag network flows with their associated context, enabling the information to be shared and acted upon within the network and end-devices. We establish the conceptual framework behind this approach and present some application scenarios, particularly in mobility and QoS adaptation.

1 Introduction

Today we find ourselves almost completely blanketed by a plethora of communication networks including those provided by different mobile phone services, privately-owned and public "hotspot" wireless LANs, personal area networks that use Bluetooth, as well as satellite-based mobile communication and Internet services. The diversity of these networks is only rivaled by variety of the features and characteristics of the mobile devices and applications that run on them.

As we shift from one activity to the next, we find it increasingly inconvenient, if not outright difficult, to continuously and consciously adapt to the different devices and connectivity modes appropriate to our activities, as well as to the constant changes in network characteristics and conditions. One way to mitigate this is by designing minimally-distracting [1]

Please use the following format when citing this chapter:

Ocampo, R., Galis, A., De Meer, H. and Todd, C., 2007, in IFIP International Federation for Information Processing, Volume 229, Network Control and Engineering for QoS, Security, and Mobility, IV, ed. Gaïti, D., (Boston: Springer), pp. 257–268.

networks that automatically adapt to changes in conditions as well as users' activities with little or no user intervention, that is, by developing context-aware networks.

In this paper, we discuss the conceptual framework for flow context tagging as an approach in building context-aware networks. We use the term "flow" to refer to distinguishable streams of related datagrams resulting from the activity of a single entity [2], although we adopt a more inclusive, end-to-end view often attributed to sessions. In our approach, flows across the network are tagged with context information, enabling network devices and end-hosts to gain more information about the flow than what would normally be provided by the individual packet headers or obtained only through stateful inspection of the flow at higher layers.

In some of our recent work we mentioned that flow context may be used within the network to trigger adaptation, personalized services, network-wide (rather than flow-directed) management actions, long-term collection of management information and knowledge-based network management. In this paper however we complement those broad usage classes with discussions on possible application areas, with particular focus on mobility and QoS, while outlining future applications in intelligent flow classification and management, overlay routing and content delivery, and in the control of malicious flows.

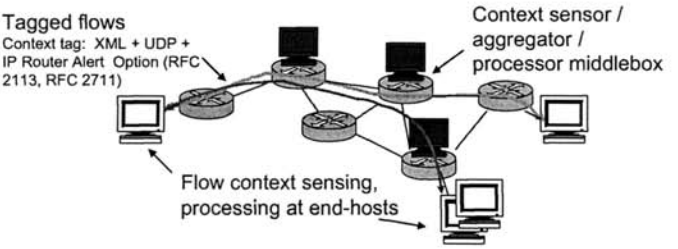
2 The Context of a Flow

Our use of the term context has its origins from the domain of pervasive and ubiquitous computing. Dey, Salber and Abowd define context as "any information that can be used to characterize the situation of entities ... that are considered relevant to the interaction between a user and an application, including the user and the applications themselves" [3]. In the human-computer interaction (HCI) and ubiquitous computing domains the particular entities of interest are usually the user or the application. For example, context has been defined as the location and identities of nearby people and objects relevant to an application [4]; the elements of the user's environment that a computer may know about [5]; or the state, situation and surroundings of the user and her devices [6]. In our case, however, we are primarily interested in the interaction between users and the network; thus the entity of interest from our point of view is the network flow, as it is the physical (or electronic) embodiment of the user's interaction with the network.

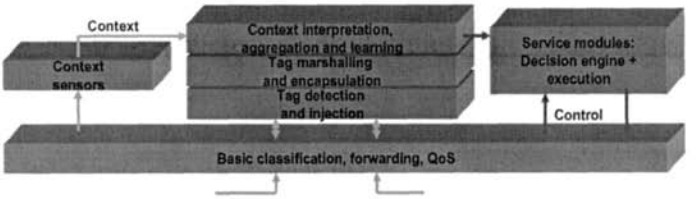
We define the context of a network flow as any information that can be used to characterize its situation, including information pertaining to other entities and circumstances that give rise to or accompany its generation at the source, affect its transmission through the network, and influence its use at its destination. This includes not only the intrinsic, low-level characteristics of a flow, but also the nature of the applications, devices, and the activities, intentions, preferences and identities of the users that produce or consume the flow.

3 Tagging Flows with Context

Recently we outlined the mechanics and architecture of our approach [7,8]. For completeness, we review some of its key elements, which are also illustrated in Fig. 1.



(a) Typical network deployment



(b) Context tag processing stack

Fig. 1. Functional components of flow context tagging approach

Architecture Context sensing functionality is implemented primarily at end-hosts, where there is rich context information about user and application activity and device capability. Network devices and middleboxes may

also perform context sensing either through flow inspection or by processing context information from other sources. Context tags are then assembled and injected along the path of the flow and are intercepted and processed by devices along the flow's path. In some cases the context tags may trigger a control or management action, a service, or an adaptation function within a downstream network device such as a router. End-hosts may also process context tags.

Tag structure Tags are formatted using Extensible Markup Language (XML) and transported within UDP datagrams. XML provides an extensible way to represent context information, and allows the formal specification of languages governing context interpretation. The IP packet header contains the IP Router Alert Option as described in RFC 2113 and RFC 2711.

Tag aggregation Tag processing may also result in the aggregation of information coming from multiple tags accumulated over time, or from multiple flows, resulting in higher-level context information that provides a more complete contextual description of a single flow or a flow aggregate (macroflow). Tag aggregation also enhances scalability by reducing information overload, as the network has the option to process and maintain state for progressively fewer tags with higher-level semantic content as flows approach a network's core.

Ontology An ontology that formally encodes the relationship and properties of the entities within context tags allows the development of a common vocabulary between context producers and consumers within the network and promotes interoperability across domains. Declarative semantics within the ontology facilitate the use of reasoning within the tag aggregation process, and provide a means by which (macro-) flow characteristics and requirements may be derived using inference.

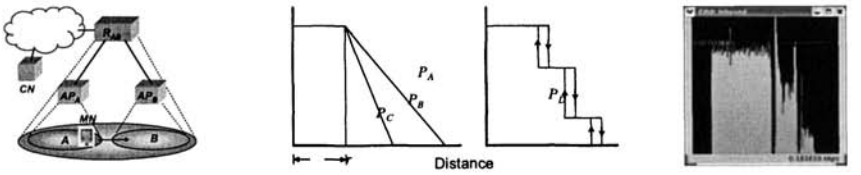
Incremental deployment Context sensing functionality may be added to end-hosts or incrementally on network nodes such as routers, or dedicated boxes may be inserted within the network in order to inspect flows and inject context tags. For nodes that will provide context-triggered services, the service modules and the core router functionalities (classification, forwarding, QoS) do not necessarily have to be closely coupled; the context-related and adaptation functions could reside on a separate device "bolted" onto a conventional router, and SNMP may be used to effect service execution.

4 QoS and Mobility Adaptation

4.1 Mobility adaptation

Context tags may be used as a means by which mobile hosts may announce information on their geographic location or movement to other nodes, or for geographic routing. They may also be used to share other useful information such as device capabilities or battery levels with mobile peers. A flow that indicates a transmitting host with critically low battery levels, for example, may be given priority through the network in order to avoid a costly retransmission.

In this section we describe an application of context tagging to mobility as follows: a mobile node MN (Fig. 2a) requests a multimedia UDP stream from server corresponding node CN. The stream is sent by CN to MN, the latter accessing the network via access point AP_A. MN then moves from the coverage area A of AP_A to coverage area B of AP_B. As it moves, `FlowSourceGeographicLocation` updates in its context tags allow nodes along the path to directly infer its location and movement.



(a) Reference diagram

(b) Adaptation profiles

(c) Actual response

Fig. 2. Reference network, adaptation profiles and node response

As MN moves towards coverage area B, router R_{AB} detects through the MN's context tags that it is starting to move within proximity of the coverage area of AP_B. Through a mechanism we call *speculative multicast* the return stream is also sent to AP_B, even while MN has not yet formally handed over to AP_B. Policy settings on the access point AP_B determine whether streams received via speculative multicast are actually sent over the radio interface, or simply cached within the access point. The advantage of having this mechanism is that it offsets the effects of discovery

time, or the amount of time before a mobile discovers that it has moved into or out of a new wireless overlay, which often dominates handoff latency [9]. The moment MN joins the new AP it has immediate access to portions of the stream that would otherwise be lost due to handoff latency.

In addition, the type and characteristics of the flow content, e.g. whether its bitrate can be modified either through distillation [10] or by dropping layers [11] are also contained in the context tag. If bitrate modification is possible, an *adaptation profile* (Fig. 2b) specifies the target rate in relation to the probability that it will join a specific access point. We define the *miss penalty* as the number of bits speculatively multicast to an access point that are not actually received by the mobile host because it has not yet joined an access point. A profile P_B with a relatively steep slope reflects a more conservative policy than P_A, as it allows high speculative multicast rates only when a mobile node has a high probability of joining the access point. Consequently it has a lower expected miss penalty. The degenerate case P_C only allows streams to be sent if the AP is actually servicing the MN, which is the behavior seen in many schemes today.

Profile P_D implements a more pragmatic stepwise adaptation, rather than the linear and somewhat idealized profiles shown by P_A and P_B. Figure 2c shows the actual inbound bandwidth on an access point implementing such a profile. The maximum bandwidth received by the access point in this example is when the mobile node is within the coverage area. Even as the mobile node leaves the coverage area, the access point still receives the video stream from the router, although the bandwidth has been reduced by transcoding.

4.2 Implicit QoS signaling

To provide QoS in networks, end-hosts are often expected to either explicitly signal their QoS requirements and undertake resource reservation, or to have sufficient knowledge about the underlying QoS model in order to map application flows to existing QoS classes. However, in [8] we described a scenario that used context tags in implicitly signaling the QoS characteristics and requirements of network flows. We proposed that flow context may be used to: (1) decouple end-hosts and applications from the underlying domain-specific QoS model by providing high-level flow descriptors that can be mapped to a domain's specific QoS mechanisms, (2) provide or expose additional information about the flow to the network in an explicit way to facilitate flow classification for QoS purposes, (3) trigger an appropriate QoS adaptation response on the flow, and (4) identify

and label suspicious and malicious flows, or those that are in violation of QoS contracts. This section describes another experiment we conducted to demonstrate some of these aspects.

Simple proof-of-concept We transmitted a video stream with a natural bit rate of approximately 850 kbps, which the network had to reduce to a target 500 kbps based on an adaptation profile similar to those shown in Fig. 2b. A simple (and perhaps default) way for the network to achieve this would be to impose a hard limit on the allowable bit rate of this flow, as shown in Fig. 3, *left*. While this might be acceptable for elastic flows, such context-unaware QoS adaptation might be unsuitable for packet loss- or delay-sensitive traffic such as video, as shown. On the other hand, injecting the appropriate context tags informs downstream adaptors that the flow content may be modified through transcoding. A transcoding adaptor is triggered with an output bitrate parameter setting that corresponds to the 500 kbps target traffic rate. The traffic profile produced by this adaptation and the corresponding video quality are shown in Fig. 3, *center*. The traffic profile shows some "spikes," artifacts of the transcoding scheme used. In order to prevent these, a combination of the bandwidth limiting and transcoding adaptation strategies using adaptor composition results in the traffic profile and video quality shown in Fig. 3, *right*. Occasional and minor degradation of video quality was observed, but the overall quality experienced during the experiment was acceptable.

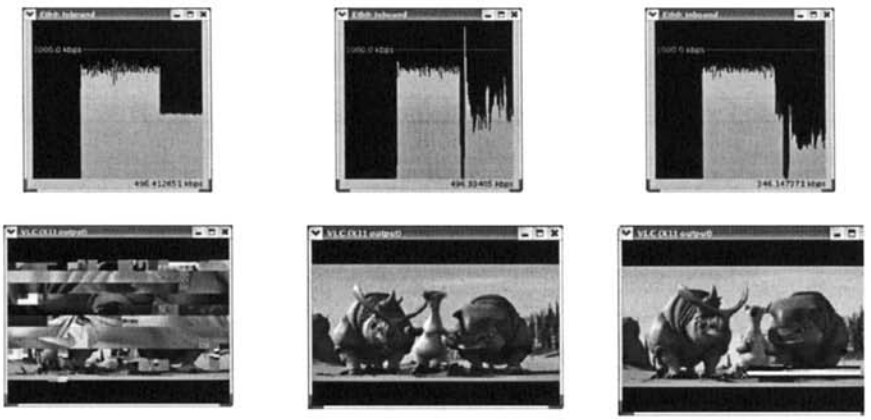


Fig. 3. QoS adaptation strategies. *Left*: Context-unaware. *Center*: Context-aware. *Right*: Context-aware, with adaptor composition

Although not shown here, the function in Fig. 3, right would not be realized properly if band-limiting the flow preceded transcoding its content. This problem may be handled by explicitly stating the interoperability parameters of each adaptor module (or service module as a generalization), specifying the input and output conditions necessary to cascade or compose them [12].

This simple experiment shows how context tags may be used to signal QoS requirements and acceptable QoS adaptation strategies. In this example we have shown implicit signaling, as the sending host had no prior knowledge of the QoS adaptation model existing within the network. In this case it was up to the network to decide which adaptation strategies were appropriate, given the flow's context and the network's QoS goals.

5 Other application areas

Intelligent flow classification and management There are instances where information needed for flow classification such as network addresses or transport-layer port numbers are modified, such as when network- or port address translation (NAT/PAT) are used, or when traffic is tunnelled within well-known protocols such as the Hypertext Transfer Protocol (HTTP) as a stealth technique or as a means to bypass firewalls. As an alternative to each network node performing costly stateful inspection, the flow's context may be sensed either at end-hosts or by dedicated middleboxes, and this information could be shared throughout the flow path so that network may properly classify the flow.

Overlay routing and content delivery Requests for content streams and the corresponding delivered content may be classified and routed through a network based on context tags. In the case of multimedia or real-time streams, the flow that contains the content request may contain a description of QoS, cost, security or reliability requirements that the underlying network may use as basis for a routing decision or to map the flow to an appropriate overlay. In the reverse direction, the flow containing the content to be delivered may contain a description of both the requirements of the requestor and the characteristics of the content, again for routing or overlay mapping purposes.

Mitigating attacks and controlling malicious flows A node equipped with sensors that can detect distributed denial of service (DDoS) attacks or worms propagating through the network may inject a context tag

in the reverse path so that upstream nodes may suppress the inbound flows. The tag may contain a description of the malicious flow that upstream nodes may use as a pattern to detect and suppress subsequent attacks, ultimately at or near their sources. This technique may also be explored as a means of controlling spammed email and mitigating its impact on the network. Spam is typically marked, classified or discarded at the receiving end; by the time it has reached its destination, it has already wasted a significant portion of the network's bandwidth. Context tags would enable a network-level response that may also propagate all the way near the source of the spam traffic. This is especially significant in suppressing continuous spam traffic that originates from hijacked hosts on broadband networks.

6 Related Work

Our concept of context-tagged flows seems synonymous with or related to the concepts of "context-aware communications" and "context-sensitive communications." Henricksen et al. use the term context-aware communication to refer to the use of context in communication applications; however, the applications they cited as examples use "communications" in a sense that often pertains to the direct interaction between humans or at the application layer [13]. In addition, the primary consumers of context in these cases were end-applications or even humans, rather than network devices. Context-tagged flows seem to be more related to the general concept of context-sensitive communications (CSCs) in reference to context-triggered, impromptu and possibly short-lived interactions between applications in ubiquitous computing environments [14,15]. CSCs are also defined as a type of communication where channels are established between devices based on some specific contexts, and are used for context dissemination to network entities [16]. Context-tagged flows also share these properties of CSCs and are likewise used for context dissemination to network entities and end-hosts. However, tagged flows provide a very specific approach to the dissemination of context within the network, and present a different persistence model.

Our approach shares some architectural and conceptual similarities with COPS (Checking, Observing and Protecting Services), where middleboxes called iBoxes (Inspection-And-Action Boxes) perform deep packet inspection in order to identify and segregate traffic into good, bad and suspicious classes [17]. Annotation Labels are inserted into packets and used as basis within the network whether a packet is to be forwarded normally, slowed, or dropped. COPS seems to be focused primarily on network protection

and QoS, and has a limited notion of context. We believe that our broader view of flow context presents a more general framework and may lead to a wider class of novel and useful applications.

In mobile applications, handoff latency can also be reduced through doublecasting [9], whereby packets within a wireless overlay are simultaneously sent to another base station belonging to another overlay in the network. This however is intended for vertical handoffs between overlays that use different network technologies, as it assumes that the mobile host simultaneously receives on different network interfaces. In multicast-based mobility (M&M) [18] a mobile node is assigned a multicast address and throughout its movement, joins a multicast tree through locations it visits, but only after it has actually moved to the new location. In contrast, in our approach, the inbound stream may actually be sent to the base stations or access points covering the mobile host's next possible locations, even before it has actually moved there.

In the QoS application domain, HQML [19] is an XML-based hierarchical QoS markup language that allows applications to signal QoS characteristics and requirements to end-applications and network elements called QoS proxies. However, it is focused specifically on QoS and on Web applications, and is not designed as a general mechanism for making other types of context information available to network nodes. The Session Description Protocol (SDP) [20] describes multimedia sessions using a short textual description that includes information on media, protocols, codec formats, timing and transport information, while Multipurpose Internet Mail Extensions (MIME) [21] provide high-level type descriptions for different content types such as text, images, video, audio or application-specific data in message streams. Unlike context tags, these schemes deliver flow or session context to end-hosts rather than network nodes, and are limited to very specific application domains. However, the formats and types used in SDP and MIME messages may be reused to describe flows in a high-level way within context tags and our flow context ontology.

7 Conclusion and Future Work

Context awareness can help build minimally-distracting networks by enabling them to automatically adapt to the user's activities, computational environment, and network conditions. Our approach in building context-aware networks considers network flows as entities of interest, and uses context information that encodes the nature, state, requirements and other relevant information that describes these flows. By tagging flows with con-

text, we provide a means by which network devices as well as end-applications can adapt to them, or cause long-term management actions to be performed in an intelligent way.

We have described some application areas where our context-tagging technique may be applied, such as in mobility and moving networks, QoS, intelligent flow classification and management, overlay routing and content delivery, and in the control of malicious flows. We are conducting a more rigorous validation and performance evaluation of our approach in these application areas, and expect to uncover more possible applications in the course of our work.

Acknowledgment. This paper describes work partially undertaken in the context of the E-NEXT - IST FP6-506869 project, which is partially funded by the Commission of the European Union. The views contained herein are those of the authors and do not necessarily represent those of E-NEXT. Roel Ocampo acknowledges support from the Doctoral Fellowship Program of the University of the Philippines.

References

1. M. Satyanarayanan. Pervasive Computing: Vision and Challenges. IEEE Personal Communications, 8(4), August 2001
2. R. Braden, D. Clark, and S. Shenker. Integrated Services Architecture in the Internet: an Overview. Request for Comments 1633, June 1994.
3. A. K. Dey, D. Salber, and G. D. Abowd. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Human-Computer Interaction (HCI) Journal, 16 (2-4), 2001
4. B. Schilit and M. Theimer. Disseminating Active Map Information to Mobile Hosts. IEEE Network, 8 (5), September 1994
5. P. Brown, J. Bovey and X. Chen. Context-Aware Applications: From the Laboratory to the Marketplace. IEEE Personal Communications,4 (5), October 1997
6. A. Schmidt, M. Beigl, and H. Gellersen. There is More to Context Than Location. Computers and Graphics Journal, 23 (6), December 1999
7. R. Ocampo, A. Galis and C. Todd. Triggering Network Services Through Context-Tagged Flows. Proceedings of the Second International Workshop on Active and Programmable Grid Architectures and Components (APGAC'05), Atlanta, Georgia, May 2005
8. R. Ocampo, A. Galis, H. De Meer, and C. Todd. Implicit Flow QoS Signaling Using Semantic-Rich Context Tags. Proceedings of the 13th International Workshop on Quality of Service (IWQoS 2005), Passau, Germany, June 2005

9. M. Stemm and R. Katz. Vertical Handoffs in Wireless Overlay Networks. Mobile Networks and Applications. Special Issue: Mobile Networking in the Internet, 3 (4), 1999
10. A. Fox, S. D. Gribble, Y. Chawathe and E. A. Brewer. Adapting to Network and Client Variation Using Active Proxies: Lessons and Perspectives. Proc. 16th Intl. Symposium on Operating Systems Principles (SOSP-16), France, October 1997
11. S. McCanne, V. Jacobsen and M. Vetterli. Receiver-Driven Layered Multicast. Proceedings of the ACM Sigcomm '96 Conference, August 1996
12. M. Yarvis, P. Reiher and G. Popek. Conductor: A Framework for Distributed Adaptation. Proc. 7th Workshop on Hot Topics in Operating Systems, March 1999
13. K. Henricksen, J. Indulska and A. Rakotonirainy. Modeling Context Information in Pervasive Computing Systems. Proceedings of the First International Conference on Pervasive Computing, Zurich, Switzerland, August 2002.
14. S. Yau and F. Karim. An Adaptive Middleware for Context-Sensitive Communication for Real-Time Applications in Ubiquitous Computing Environments. Real-Time Systems, 26 (1), 2004
15. M. Khedr and A. Karmouch. Exploiting Agents and SIP for Smart Context Level Agreements. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Canada, August 2003
16. A. Karmouch, A. Galis, R. Giaffreda, T. Kanter, A. Jonsson, A. Karlsson, R. Glitho, M. Smirnov, M. Kleis, C. Reichert, A. Tan, M. Khedr, N. Samaan, H. Laamanen, M. El Barachi and J. Dang. Contextware Research Challenges in Ambient Networks. 1st International Workshop on Mobility Aware Technologies and Applications, October 2004
17. R. Katz, G. Porter, S. Shenker, I. Stoica and M. Tsai. COPS: Quality of Service vs. Any Service at All. Proceedings of the 13th International Workship on Quality of Service (IWQoS 2005), Passau, Germany, June 2005
18. A. Helmy, M. Jaseemuddin and G. Bhaskara. Multicast-based Mobility: A Novel Architecture for Efficient Micro-Mobility. IEEE Journal on Selected Areas in Communications (JSAC), Special Issue on All-IP Wireless Networks, May 2004
19. X. Gu, K. Nahrstedt, W. Yuan, D. Wichadakul, and D. Xu. An XML-Based Quality of Service Enabling Language for the Web. Journal of Visual Language and Computing (JVLC), Special Issue on Multimedia Languages for the Web, February 2002
20. M. Handley, V. Jacobson. SDP: Session Description Protocol. Request for Comments 2327, April 1998
21. N. Freed and N. Borenstein. Multipurpose Internet Mail Extensions Part Two: Media Types. Request for Comments 2046, November 1996