On Flexible Topology Formation in Publish-Subscribe Networks

Borislava Gajic, Janne Riihijärvi and Petri Mähönen
RWTH Aachen University, Institute for Networked Systems
Kackertstrasse 9, D-52072 Aachen, Germany
Email: {gbo, jar, pma}@inets.rwth-aachen.de

***Abstract*—In this work we analyze topology discovery and the procedure for joining the network in the information-centric context. We develop and evaluate such a network attachment procedure in an information-centric network utilizing the publish-subscribe paradigm for the data exchange. In our work the publish-subscribe concept is not only used as a communication means, but we aim at fully exploiting its characteristics for native merging of fine-grained network operations such as topology management and network connectivity establishment. Such an integration adds not only to the simplicity of the network and efficiency of the network information gathering, but includes the means for handling mobility issues. We examine the performance characteristics of the proposed solution particularly focusing on complexity and introduced message overhead. The evaluation results obtained from the testbed experiments show the outstanding performance in terms of delay, while the signaling overhead remains at a very low level.**

I. Introduction

Given an arbitrary node in the network the prerequisite for establishing the communication with the rest of the nodes is to be connected with at least one another node that can act as a bridge towards the rest of the network. The procedure of new node's joining the network commences by setting up such initial connectivity. Depending on the network and the protocol design the establishment of the primary connection with the network may be followed by a sequence of additional operations before the node completely joins the network. We refer to such a protocol for establishing a full network connection as a *network attachment*. We consider the *network attachment* as a set of operations required for initializing and maintaining the connectivity. It consists of new-node's discovery followed by corresponding modification of network states in order to address the changes in network configuration. The attachment procedure can embrace a variety of additional mechanisms, e.g., authorization and security features which are beyond the scope of this paper.

The *network attachment* in the currently predominant IP-based networks is handled on the data link layer and the network layer, using the common send-receive pattern for communication bootstrapping. In this work we are considering the novel approach in attachment procedure specially designed for information-centric networks, entirely relying on data-centric type of communication. Instead of send-receive primitives the connection bootstrapping is based on the publish-subscribe paradigm [1] .

In the present Internet the user interested in receiving some data initiates the communication process by retrieving the IP address of the data source. Having the target IP address allows the end user to directly contact the source in order to request the desired data. Therefore, the sender is given the power to drive the data transfer, leaving the receiver without direct control over the data transmission. Consequently, the misbehavior of the sender can cause various problems, e.g., denial of service and spam attacks. As opposed to the standard sender-oriented communication, in the publish-subscribe context the end users only express their interest in receiving some data by subscribing to it. On the other hand, the users declare their willingness to provide the data by publishing it. As an outcome, the end users are supplied with data from the most suitable source without knowing its identity or the address and only after explicitly expressing their interest in receiving that particular piece of information. This fundamental difference between sender-oriented and the publish-subscribe type of communication stimulated a lot of research work focused on restructuring the current IP-based Internet in order to operate on data-centric concepts. The information-centric model fits well to a wide range of already existing applications in the current Internet as well, where the users are primarily interested in the content they are willing to receive, regardless of the source from which the data is provided.

A variety of architectures have already been developed around the publish-subscribe concept appointing the content as a focus for the data transmission [2]–[6]. The PURSUIT project [7] advocates the clean-slate redesign of the present Internet architecture, establishing the new model of the Internet entirely on the publish-subscribe paradigm. Our work takes as a reference the PURSUIT architecture but being focused on data-centric communication it is adaptable to other information-centric architectures, as well. The main building blocks of PURSUIT architecture are *rendezvous*, *topology* and *forwarding* modules. After the publications and subscriptions to particular data item have been matched on the rendezvous system, the rendezvous initiates the process of generation of the optimal forwarding paths from the publisher to the subscriber by sending the request to the topology manager. Finally, the forwarding system performs the data transfer based on the delivery paths generated by the topology module. Our target is to enrich such information-centric architecture with the flexible *network attachment* protocol in order to better address the possible fluctuations in the topology. Having a *network attachment* which can provide the smooth integration of a new node in the network without affecting the usual network functionality improves the performance of the network in dynamic environments. Apart from the prime functionality of incorporating the new node into the network, the information-centric nature of underlying network offers a variety of benefits and optimization opportunities. Utilizing the hierarchical naming structure present in the PURSUIT architecture gives us the possibility of integrating the topology management and *network attachment* functionality. The procedure of attaching a new node to the network is tightly related to the dissemination and collection of network knowledge. Merging it with the nodes' naming structure allows instantaneous distribution of the updates on topology states. Such a novel approach designing the attachment protocol as a helping functionality of topology management and incorporating the naming scheme into the topology data gathering and dissemination, fully utilizes the information-centric communication paradigm.

The rest of this paper is organized as follows. In Section II we outline the the main design principles and the model description. Section III explains the implementation objectives of our attachment module. We proceed by illustrating and discussing the evaluation results of the proposed model in Section IV. In Section V we present the related work in the domain of *network attachment* protocols. Finally, in Section VI we draw the conclusions.

II. Model description

According to the PURSUIT architecture the information exchange is conveyed by means of publications and subscriptions to specific data item. The information is structured as an acyclic graph where the leaf vertices represent the individual information items. In order to facilitate data dissemination and searching mechanisms the notions of scopes and dissemination strategies are introduced. The scoping represents the grouping of related data items into a whole for easier data accessibility. The scopes are defined in a way that every information item belongs to at least one scope. Different scopes are linked together depending on their semantic relations. The dissemination strategies dictate the exact implementation of a scope by restricting its visibility to particular domains. Individual data items are denoted by the rendezvous ID (RID), which is unique in the scope to which the data item is assigned. The scope IDs are created in the similar manner where the given scope ID (SID) is unique within the super-scope to which it is linked. Therefore, the RID together with the belonging scope IDs fully and uniquely identifies the data. Similarly, every scope is uniquely marked with its SID along with the identifiers of the scopes that it is semantically linked to. Such data structure resembles the hierarchical organization of scopes and belonging sub-scopes and information items. Figure 1 illustrate the described data structure. The information item with the rendezvous ID *RID3* is fully identified by the *SID1/SID2/SID3/RID3* while the scope with the scope ID *SID2* is identified by *SID1/SID2*. The information items denoted by *SID1/SID2/RID1* and *SID1/SID2/SID3/RID1* refer to different data, the identifier *RID1* is unique for the scope it belongs to.

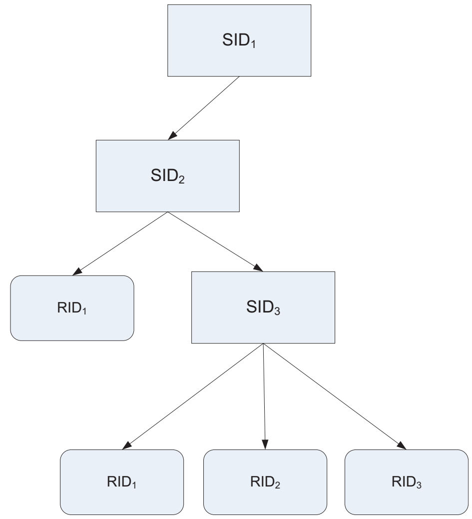

Fig. 1. The PURSUIT hierarchical data structure

Topology management module represents one of the fundamental building blocks of the information-centric architecture. Its function is to find the optimal forwarding paths from publishers towards subscribers, with respect to current network conditions. The common topology discovery and delivery path formation mechanisms such as OSPF [8] can be easily implemented using publish-subscribe pattern. Furthermore, such implementation can be applied in information-centric architecture as illustrated in our previous work [9]. Therefore, the publish-subscribe paradigm is proven to be flexible enough for supporting not only content retrieval, as already present in many Internet applications, but also more fine-grained functions such as topology discovery.

However, in order to fully utilize the strength of publish-subscribe information-centric model and to further demonstrate its flexibility in lower level network operations, we aim at merging the network attachment process with the topology discovery and intra-domain topology management. The hierarchical organization of the PURSUIT data structure provides a needed prerequisite for such integration. Furthermore, the grouping of data by means of scopes and dissemination strategies provides a powerful tool for flexible addition of new protocols. The protocol required signaling can be conveyed in specially dedicated dissemination strategy and/or scope without affecting the ordinary network functionality. Following such principles, our goal is to develop the model by which the initialization of network connectivity will entail the instantaneous dissemination and gathering of network knowledge.

In order to announce its presence each node can publish the new scope message, having the scope ID assigned according to the node's ID. Such publications can be distributed using a separate strategy. By means of dissemination strategies the data flow can be highly adapted, since the applied strategy defines the way of realizing the main functions, i.e., rendezvous, topology and forwarding. For the communication between the nodes prior attaching to the network the initial message exchange can be conveyed within the dedicated strategy, e.g., "broadcast" strategy. This type of dissemination strategy would allow nodes to communicate only with the nodes in the transmission range, similar to broadcast type of communication. Furthermore, every node incoming to the network needs to be subscribed to this strategy in order to obtain needed information for joining the network. During the network attachment phase, the node subscribed to the "broadcast" strategy will be able to receive the announcements of the nodes in the range. Thus, the joining node will gain the knowledge about its neighbors and existing links. In order to distribute this awareness the node will publish its existence in the form of new scope publication where the scope ID will be composed as neighborID/nodeID. Following such a principle, every generated scope, representing the new node in the network, carries not only the ID of the incoming node, but the information about interconnectivity, as well. In other words, the bootstrapping of connectivity is seamlessly combined with the network knowledge collection and dissemination. The topology management can collect the information about incoming nodes by subscribing to the dedicated strategy. By this means the topology management is constantly updated about the current network states.

In order to illustrate aforementioned model we give an example. Let us assume existence of only one node in the network, with the ID *AA*. In order to announce its presence in the network the node publishes the scope publication with the ID *AA*, using the "broadcast" strategy. Every new node in order to attach to the network needs to subscribe to this "broadcast" strategy. The new node with the ID *BB* will hear the message from the node *AA* and infer that there is the link between them. Then, the new node will announce its presence by publishing a new scope under the ID which is the composition of the neighbor ID and its own. Thus, in this particular case the ID *AA/BB* would represent the new node (scope). In this way such announcement will contain not only the information about the node's ID, but the information about its place in the network with respect to other nodes, as well. Such announcing procedure repeats in the same way for each incoming node, regardless of the position in the network. The nodes can be connected with more than one neighbor at the same time, which would be announced by publishing the same scope under different paths.

Finally, the topology manager will receive all messages of the type: new scope ID *AA*, *AA/BB*, *AA/DD*, *AA/BB/CC*. Thus, based on the scope IDs it can deduce the information about links and nodes existing in the network and perform required changes. The nodes' announcements can be periodic, so that the topology manager is dynamically updated in the case that the nodes join or leave the network. Figure 2 illustrates the described hierarchical name assignment in the attachment procedure.

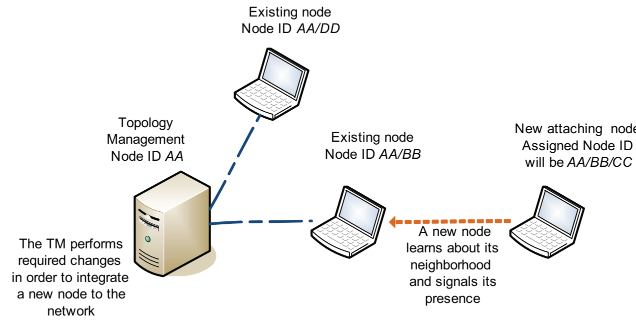

Fig. 2. An illustration of the use of hierarchical naming for dynamic topology discovery.

A. Spatially and technologically related scoping hierarchy in the context of topology data dissemination and gathering

Similar to the organization of current Internet, the information-centric systems assume the network division into the operational units equivalent to administrative domains. Although the interconnections between different network domains can have highly complex structure, the above described hierarchical data organization facilitates the establishment of semantic references among the network units. Moreover, the fine-grained partitioning within the single domain with respect to the technology used in the sub-domains can be easily delineated, as well.

In the context of information-centric networks every administrative domain is controlled by a single topology management entity. Such individual network unit can be identified by the scope ID. Furthermore, spatially smaller operational areas or technologically distinctive parts of a single domain (e.g., area covered by one access point, or the area applying one particular communication standard) can be denoted by separate scope IDs derived from the parent domain scope ID. Such a generation of the scope IDs according to the belonging scope ID can be done by utilizing various algorithms. The details about the algorithms for identifiers' generation are beyond the scope of this work. The most important asset of applying algorithmic IDs generation in naming the different parts of the single domain is in keeping the tight relations between the structural pieces.

Moreover, such mapping of distinctive spatial and technological units with hierarchical naming structure can be combined with aforementioned integrated data dissemination and gathering procedure. If during the attachment procedure for each of the joining nodes, the node ID is assigned according to its affiliation to particular spatial or technological segment of the domain, the announcement of node's existence in the network will also signal its belonging to the specific domain. Therefore, the process of network discovery attains another dimension. Apart from the process of gathering the knowledge about existing connections, the network attachment can concurrently provide the information regarding the belonging area or technology and standard used.

B. Integration of mobility solutions

The frequent changes in network topology due to the mobility of nodes can induce a significant packet loss. The methods for handling the mobility issues can place a considerable burden on the overall network operation. However, the publish-subscribe communication pattern provides a firm basis for mitigation of mobility issues due to its data-centric characteristics. The desired content can be retrieved from various locations, thus, the connection loss due to mobility does not necessarily involve inability of fetching the desired data. Furthermore, efficient mobility solutions can be incorporated within network attachment model described above. Given that the announcement messages prior attaching indirectly contain the information about node's neighborhood, the modification in the node's location due to mobility will be directly reflected in its announcements.

By changing the position, the node's neighborhood may alter. The moving node being disconnected from previous attachment points can come into the proximity of other nodes to which it was not connected before. Due to the model design the simple subscribing to the dedicated strategy during the attachment phase on the new location will result in acquiring an updated neighborhood information. The following announcement of node's existence will incorporate the updated paths to the node. Due to the periodic nature of such announcements the topology management is able to notice any change in nodes position relative to the other nodes. After the node moves, the topology manager will stop receiving announcements from the node on the old location, but the announcements from the same node on changed position will be signaled. Thus, the topology management can immediately perceive occurred modification in node's location and change the topology states accordingly.

III. Implementation objectives

The hierarchical organization of the PURSUIT functional model provides the base for utilizing the naming structure for more efficient network discovery through the seamless integration of the topology management and network attachment operations. Having the nodes' naming structure depicting also the interconnectivity in the network gives the possibility of instantaneous gathering of network knowledge upon the initialization of the attachment process. For this mode of operation the underlying architecture needs to offer the possibility of gathering the network information prior attaching, e.g., utilizing the "broadcast" strategy. Such a way of communication would allow the new-coming nodes to collect the knowledge about the current network state, based on which the attaching procedure, taking advantage of hierarchical naming, can be performed. In general, the initial information gathering can be performed in different ways depending on the underlying architecture and protocols used. Our early implementation efforts are focused on building the attachment procedure utilizing the hierarchical naming structure and entirely relying on publish-subscribe communication pattern, regardless of the method used for initial information gathering. Our attachment procedure signals the existence of a new node willing to connect to the network, and handles the necessary operation for its smooth integration. Our implementation guideline is to keep the ordinary network operation unaffected by such procedure. Furthermore, the delay, complexity and introduced signaling overhead should be minimized.

We aim at building such network attachment module as a helping functionality of the existing PURSUIT intra-domain topology management [10]. All nodes in the network can signal to the topology manager the arrival of a new node in the network by publishing the messages in dedicated strategy and scope. The attachment module, as a helping module of topology manager, supplied with such information triggers the set of operations for integrating the new node into the network. The adequate modifications in the topology management, as well as in the particular forwarding nodes are required.

The topology management maintains the igraph [11] states as the graph representation of the network. Thus, upon the arrival of the new node, the igraph states need to be updated accordingly by adding the new vertices and edges. After the igraph states have been changed, the forwarding nodes being affected by this modification need to update their states correspondingly. According to the PURSUIT architecture, all nodes are determined by the set of click modular router [12] elements. The forwarding functionality of a node is defined in the forwarding click element. This element maintains the forwarding states needed for proper data routing, as a counterpart of routing entries. Adding the new node, and thus the new links, needs to be signaled to affected forwarding elements for updating their states accordingly. Being aware of all occurred changes that the joining of a new node caused, the attachment module generates the instruction messages for changing the forwarding states. These messages contain the information about the forwarding states that need to be added to the existing forwarding configuration. The messages are published to the involved nodes using the special strategies and scopes for distribution of network notifications, to which all the nodes in the network are constantly subscribed. Once the involved node receives the instruction message it incorporates the forwarding states' update into its configuration. Along with this update the network attachment procedure is completed; the new node is integrated in the network and can perform the data exchange by means of publications and subscriptions.

IV. Evaluation results

In order to evaluate the performance of our network attachment module, we execute the set of experiments in the testbed environment. The nodes in the testbed are running the PURSUIT prototype implementation and are connected in the local area network. Our aim is to investigate the impact of attaching of a single node in terms of delay, complexity and overhead. Therefore, we perform the experiments by attaching the node in different positions in the network with respect to the topology manager node. We measure the delay due to the processing in igraph and in click forwarding elements. The processing in igraph is required for changing the network graph representation by adding the new vertices and

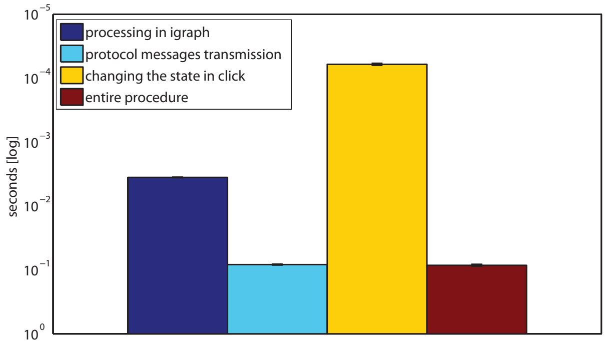

Fig. 3. The delay introduced by the attachment procedure. A new node is attaching directly to the topology management node. *Note: Due to the low delay values the y-axis is given in logarithmic scale. The higher bars indicate lower delay values.*

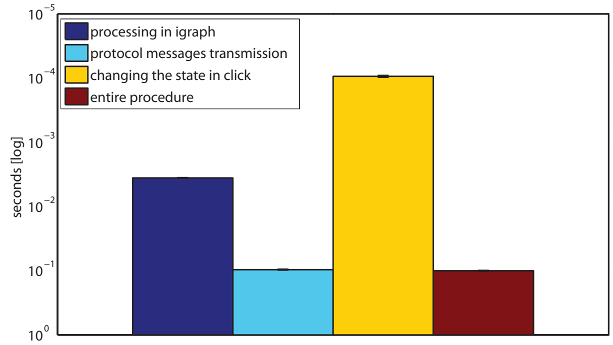

Fig. 4. The delay introduced by the attachment procedure. A new node is attaching to the node which is positioned 1 hop away from the topology management node.

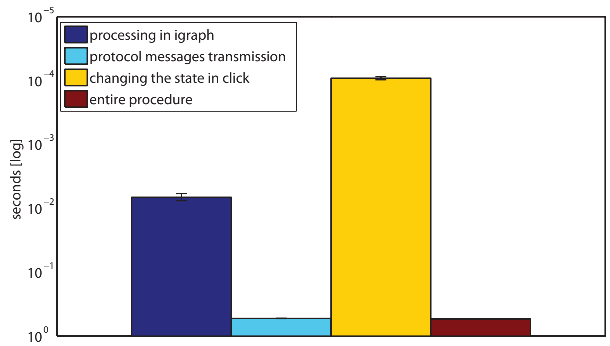

Fig. 5. The delay introduced by the attachment procedure. A new node is attaching to the node which is positioned 2 hops away from the topology management node.

edges. The modifications in the forwarding nodes are required due to the additions of the new entries in the forwarding configuration. These forwarding changes occur directly in click forwarding elements. Figure 3, Figure 4 and Figure 5 show the obtained results. Regardless of the position in which the new node is attaching, the processing requirements in igraph and forwarders remain the same. The delay introduced by performing necessary modifications has the approximately constant value regardless of the conditions under which the new node attaches. Increasing the distance of the attaching point with respect to the topology manager has as a result only the increase of the messages transmission time. The messages transmission delay is dependent on the network setup and current state of the links and is not related to the attachment operation. Figure 3, Figure 4 and Figure 5 illustrate the delay of the overall attachment procedure, as well. This delay is primarily dependent on the message transmission delay, thus, the network conditions. However, the delay caused by the network attachment required processing stays nearly constant and negligibly low (few milliseconds) regardless of the network conditions and the location of the attaching node. The results are independent of the network size, as well. This is due to the nature of our model and our experiments where we investigate the attachment of a single node to the network. The increase in the network size can have as a consequence only the increase in the message transmission delay, while the actual processing time required by attachment process remains unchanged. Furthermore, being focused on intra-domain solutions, we limit our testbed size to up to ten nodes, without affecting the generality of the results.

Another important aspect in evaluation of our network attachment module is investigation of the overhead that the overall procedure introduces. In order to accomplish the attachment process the exchange of only two messages is required. The first message signals the arrival of a new node, which serves as a trigger for starting the attachment procedure. After the adequate changes in the igraph have been made the second message is published by the network attachment towards involved nodes as an instruction for changing the forwarding states. Figure 6 shows the obtained results in terms of the message overhead. The message overhead, thus, the amount of data generated due to the attachment is constant regardless of the position of the attaching node and it includes only two above described messages.

V. Background and Related Work

The attachment procedure in present IP-based networks involves a large set of authentication and authorization mechanisms prior the final joining of the node in the network. The Extensive Authentication Protocol EAP [13] has been used as a basis for defining the 802.1X [14] protocols standard for connecting the node to the wireless or wired network. The 802.11X establishes the access control on the port level. On the other hand, in the 3G/UMTS networks the AKA-Authentication and Key Agreement [15] challenge-response based protocol is used. The point to point protocol PPP [16], the common protocol for communicating over point-to-point links, uses the Challenge Handshake Authentication Protocol (CHAP) [17] for the authentication. In general, during the network attachment procedure certain steps, e.g., network detec-

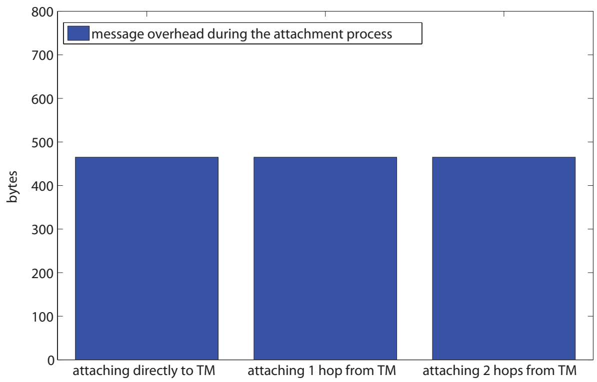

Fig. 6. The message overhead due to the network attachment procedure. The new node is attaching on different locations in the network with respect to the topology manager.

tion, authentication, address assignment, need to be performed according to the protocol design. In currently predominant networks the required steps are mostly related to the layer two and layer three operation. There has been a lot of research effort already invested in making the network attachment procedure more efficient by, e.g., minimizing the signaling overhead or cryptographic computation requirements. The problem of network attachment optimization can be broken down to the optimization of separate steps required by the overall procedure or it can be considered as a whole [18].

The network attachment in publish-subscribe systems has been considered in the scope of publish-subscribe projects mainly focusing on the authentication and security aspects [19]. The work presents the network joining solutions based on the content-based principles. The authors address the emerged security challenges due to the used communication pattern and propose certain solution directives. However, the attachment procedure is observed as an isolated process constrained mainly by security issues.

Our work gives a novel dimension to the attachment process in publish-subscribe networks by directly utilizing the benefits of data-centric communication type and particular network naming scheme. We argue on merging the topology management functionalities with the attachment protocol and thus, the smooth integration between network information dissemination and gathering. Although strongly focused on utilization of information-centric aspects in the attachment procedure, we preserve the certain level of generality. Therefore, different security schemes are applicable in addition. However, security related details are beyond the scope of this paper.

VI. Conclusions

In this work we presented a novel design for network attachment in information-centric network. The model fully utilizes the benefits offered by the content-centric communication paradigm by seamlessly integrating the network discovery with the topology creation process. Such a model can be used to facilitate the dissemination of spatial and technology specific information. Furthermore, the model incorporates the means for mitigation of mobility issues, as well. Our implementation of the attachment process shows excellent results in terms of introduced processing delay and signaling overhead. The performed experiments demonstrate the flexibility of publish-subscribe paradigm suitable not only for the content retrieval but also for more sensitive network operations, such as network discovery and topology formation. For further evaluation of proposed model different additional features of the underlying architecture and prototype implementation can be taken into consideration. Our current work in progress aims at full integration of our attachment process with the PURSUIT prototype implementation and its release as an open source.

Acknowledgements

This work was in part financially supported by RWTH Aachen University through UMIC research centre and European Union (PURSUIT project).

References

[1] R. G. P.Th. Eugster, P. A. Felber and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Computing Surveys*, vol. 35, pp. 114–131, June 2003.
[2] T. Koponen, M. Chawla, B. Chun, A. Ermolinskiy, K. Kim, S. Shenker, and I. Stoica, "A Data Oriented (and Beyond) Network Architecture," *In ACM SIGCOMM, Computer Communication Review*, vol. 37, no. 4, pp. 181–192, October 2007.
[3] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking Named Content," *Proceedings of the 5th international conference on Emerging networking experiments and technologies, ACM CoNEXT*, pp. 1–12, 2009.
[4] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, I. Stoica, and S.Shenker, "ROFL: Routing on Flat Labels," *In Proceeding of the ACM SIGCOMM workshop on future directions in network architecture*, vol. 36, no. 4, September 2006.
[5] "PSIRP-Publish Subscribe Internet Routing Paradigm," http://www.psirp.org/, [visited on 05.11.2011.].
[6] "Named Data Networking (NDN) Project," http://www.named-data.net/, [visited on 02.11.2011.].
[7] "EU FP7 PURSUIT project," http://www.fp7-pursuit.eu/, [visited on 05.11.2011.].
[8] J. Moy, "OSPF Version 2," RFC 2328, 1998.
[9] B. Gajic, J. Riihijärvi, and P. Mähönen, "Intra-Domain Topology Manager for Publish-Subscribe Networks ," *18th International Conference on Telecommunications ICT 2011*, May 2011.
[10] "The prototype implementation of PURSUIT architecture," https://github.com/georgeparisis/blackadder, [visited on 06.11.2011.].
[11] G. Csardi and T. Nepusz, "The igraph software package for complex network research," *InterJournal*, vol. Complex Systems, p. 1695, 2006. [Online]. Available: http://igraph.sf.net
[12] "The Click Modular Router Project," http://read.cs.ucla.edu/click/, [visited on 06.11.2011.].
[13] D. S. B. Aboba and P. Eronen, "Extensible Authentication Protocol (EAP) Key Management Framework," RFC 5247, 2008.
[14] T. Jeffree et al, "IEEE Standard 802.1X-2004: Port-based network access control," 2004.
[15] V. T. A. Niemi, J. Arkko, "Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)," RFC 3310, 2002.
[16] W.Simpson, "The point-to-point protocol (PPP)," RFC 1661, 1994.
[17] W. Simpson, "PPP Challenge Handshake Authentication Protocol (CHAP)," RFC 1996, 1994.
[18] J. Arkko, P. Eronen, H. Tschofenig, S. Heikkinen, and A. Prasad, "NAP - secure and efficient network access protocol," *6th International Workshop on Applications and Services in Wireless Networks*, 2006.
[19] J. Kjällman, "Attachment to a Native Publish/Subscribe Network," *The Publish-Subscribe Internet Routing Paradigm (PSIRP) EU FP7 project Deliverable*, 2009.