Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 12, December 2013, pg.224 – 228

RESEARCH ARTICLE

Self-Adaptive Context Data Distribution Infrastructures for Mobile Systems

C.Priyanka[1], Dr.A.Kathirvel[2], R.Sangeetha[3]

[1]Dept of IT, Vivekanandha College of Engineering for Women, Tamilnadu, India

[2]Professor & Head / IT, Vivekanandha College of Engineering for Women, Tamilnadu, India

[3]Asst.Prof / IT, Vivekanandha College of Engineering for Women, Tamilnadu, India

[1] priyankamtech21@gmail.com; [2] kathirvel@vcew.ac.in; [3] sgee89@gmail.com

Abstract-Context awareness is a property of mobile devices that is defined complementary to location awareness. It is essential to leverage the deployment of novel services in mobile systems. It allows dynamic adaptation of services according to the current execution conditions. Currently it suffers from ineffective context data delivery mechanisms, which introduce extreme overhead over bandwidth constrain wireless fixed infrastructures. In heterogeneous mobile devices the situation is even worse with reduced processing power and low bandwidth wireless connections. In existing reactive replication does not properly work if all close MNs have their caches full. To overcome this problem in this work we presents a new data caching algorithm that exploits peculiar aspects of context distribution for limited data lifetime and interests similarity between nodes in physical distance, to appropriately select the data to evict when necessary. We proposed a middleware called Context Data Distribution Infrastructures (CDDIs), capable of delivering context data to all interested devices, while hiding communication failures, infrastructure heterogeneity, and resource constraints. We propose a new caching algorithm, called Adaptive Context-aware Data Caching (ACDC) with a quantitative comparison of reactive and proactive strategies is proposed, specifically in terms of maximum and average resource usage. This gives insight in choosing the best strategy when designing a system, considering constraints of available bandwidth and storage for heterogeneous wireless modems and data caching techniques to effectively.

Keywords: Mobile Cellular Network, MANTET, Data Catching, Data Distribution.

I. INTRODUCTION

A cellular network or mobile network is a wireless network distributed over land areas called cells, each served by at least one fixed-location transceiver, known as a cell site or base station. In a cellular network, each cell uses a different set of frequencies from neighboring cells, to avoid interference and provide guaranteed bandwidth within each cell. When joined together these cells provide radio coverage over a wide geographic area. This enables a large number of portable transceivers (e.g., mobile phones, pagers, etc.) to communicate with each other and with fixed transceivers and telephones anywhere in the network, via base stations, even if some of the transceivers are moving through more than one cell during transmission. These connections may be able to utilize the wireless bandwidth more efficiently. The aggregated throughput in a WMN with multiple parallel TCP connections simultaneously transmitting TCP streams. The use of multiple parallel TCP connections between the transmitter and receiver that are multiple hops away can better utilize the wireless bandwidth and boost the aggregated throughput. Parallel connections between any two nodes could be a potential solution for transmitting a large amount of data at high speed from one client to one server through the wireless mesh network. TCP tuning techniques such as the use of parallel streams and dynamic adjustment of the advertised window based on the measured behavior need to be enabled in commercial wireless networking. The popularity of wireless devices and the increasing availability of heterogeneous wireless infrastructures, spanning and Bluetooth to cellular 3G and beyond, are stimulating new service provisioning scenarios. A growing number of users require any-time and any-where access to their Internet services, such as email, printing, Voice over IP, social computing, and many others more, and while moving across different wireless infrastructures. Context awareness is essential to leverage the deployment of novel services in mobile systems because it allows dynamic adaptation of services according to the current execution conditions such as computational resources, people in physical proximity, user preferences, and so on. Unfortunately, notwithstanding the importance of those new scenarios, we currently witness a quite limited diffusion of real context-aware services that in our opinion mainly stems from some important management issues not solved yet. While researchers mostly devoted to high-level issues, such as context data modeling, reasoning, and delivery to running services, the core problem of context data distribution in large scale mobile systems has been rather neglected so far. The situation is even worse for systems that include resource- constrained and heterogeneous mobile devices, with reduced processing power and low bandwidth wireless connections.

To solve this ACDC caching is proposed but reactive replication does not properly work if all close MNs have their caches full. In this work we present a middleware called Context Data Distribution Infrastructures (CDDIs) to deliver data's to hide communication failures, infrastructure heterogeneity, and resource constraints. The Context Data Management (CDM), which provides all mechanisms needed to represent, store, and cache/ replicate data in the mobile system. To improve system scalability and reduce context access times, we claim CDM should exploit complex storage architectures that include both fixed physical servers and mobile devices; in particular, CDM has to efficiently and effectively organize data storage into the system. We consider a two-layer distributed architecture: the fixed infrastructure features Fixed Nodes (FNs) that ensure context storage and availability, while the mobile one includes Mobile Nodes (MNs) that build local ad -hoc networks to share context data. Second, the CDM should enforce locality principles to reduce context visibility. We proposed a data catching algorithm called Adaptive Context-aware Data Caching (ACDC) to catch data efficiently in heterogeneous networks it works on limited data lifetime. it maintains a history H of data access descriptors, namely (data, access time) pairs, and, for each data, it combines 1) the time period between the oldest and the newest access recorded in H; 2) the number of accesses recorded in H; and 3) the remaining data lifetime, to adapt cache content according to the current access pattern. To solve reactive replication problem a quantitative comparison of reactive and proactive strategies is designed, specifically in terms of maximum and average resource usage. This gives insight in choosing the best strategy when designing a system, considering constraints of available bandwidth and storage.

II. SYSTEM MODEL

I. Network Model

The CDM should take advantage of all heterogeneous wireless modes, either infrastructure-based or adhoc, to enhance scalability. While wireless infrastructures grant context availability in larger served places, ad –hoc communications enable direct CDM distribution between devices. That is even more important when close mobile nodes are interested in the same context data, since the CDM can route them with no additional traffic on the fixed infrastructure. Hence, the CDM should exploit decentralized solutions to store and route context data into the system.

II. Context Data routing

To address context routing between sources and sinks we adopt a query-based approach to have more predictable performance and to limit runtime traffic. Also, we use distributed hierarchical architectures to foster system scalability and we exploit locality principles to find context data as close as possible to the requesting MN. Every time a source produces a new context data, the default policy is to relay it to the FN in charge of the current physical place, so to ensure persistency and availability in the system. Context queries, instead, capture the needs of context sinks and build distribution paths used by the system for data dissemination. For the sake of infrastructure offloading, each MN first distributes context queries on the same level to possibly retrieve required data on neighbors; then, only if needed, it also routes context queries to the FN. Each node that receives a query matches it against locally stored data and creates a response if a positive match occurs; a context response is then routed back to the requiring sink by using a hop-by-hop solution driven by context queries. We split context data retrieval in two phases. First, the MN gathers matching data from neighbor MNs through ad-hoc links; then, for the sake of completeness, it sends the query to the associated FN that, in its turn, can distribute the query to peer FNs. Of course, to prevent useless traffic, it is important to avoid that the FNs resend the same data the MN has already collected from peers. Hence, each query carries a Bloom filter, a space-efficient data structure for membership tests, to prevent the retransmission of already collected data.

III. Context aware data catching

ACDC adopts original technical solutions to meet CDM requirements. Because we need to take care of both local and distributed perspectives, it has both a local and a distributed nature. For the local part, ACDC adaptively tailors data ranking according to the current access pattern main goal is to adapt cached data depending on requests emitted by both local and remote sinks, so to increase the number of data successfully routed on ad -hoc links. Toward that goal, it maintains a history H of data access descriptors, namely pairs, and, for each data, it combines 1) the time period between the oldest and the newest access recorded in H; 2) the number of accesses recorded in H; and 3) the remaining data lifetime, to adapt cache content according to the current access pattern. ACDC calculates linear correlation (called correlation index) between 1) the time period among the oldest and the newest access recorded in H for a specific data; and 2) the number of accesses recorded in H for that data. To solve reactive replication problem we use specifically in terms of maximum and average resource usage. This gives insight in choosing the best strategy when designing a system, considering constraints of available bandwidth and storage. Consider a single replicated object whose lifetime is to be determined. A node can hold exactly one replica. There are $n \geq 2$ initial nodes, where n depends on, or is a measure of, the available storage. We base our models on Markova hypotheses, according to which the distributions of failure times for reactive and proactive are exponential with parameter λ. The distributions of repair times are also exponential for reactive, with parameter μ. However, in a proactive system, replications are periodic, occurring once every interval of time of length, and is the replication rate. When select values for λ and μ. When the access pattern is uniform, the history H contains access descriptors for a large set of data: each data has very few accesses descriptors in H, and that number is mostly constant between data. In our CDM, each MN periodically disseminates to its one-hop neighbors lightweight summaries of its cache: each neighbor cache summary contains the number of cached data, the maximum cache size, and a compact representation of cached data. Finally, to select the data to evict, ACDC combines together local and remote rank values and computes a utility value for each data.

IV. Performance measure

In this module the performance of the proposed algorithm is evaluated. Performance metrics are utilized in the simulations for performance comparison:

Average Retrieval Time: the average retrieval time and the percentage of satisfied requests with caching policies in {FIFO, LFU, LRU, ACDC} and HTTL in {1, 2, 3}.
Data Production Rates: The average retrieval times and the percentage of satisfied queries with data FL.
Cache Size: The same set of performance indicators used in previous sections. With larger MN cache sizes, the differences between the five caching algorithms tend to become very small.
Node Mobility: the same performance indicators of previous tests with different ranges of node speed.

III. CONCLUSION

This paper has presented our original solution for efficient context data distribution, by stressing our principal design guidelines, and highlighting how the usage of different wireless mode and distributed context caching can deeply improve CDM efficiency. We have proposed our novel ACDC replacement policy with a quantitative comparison of reactive and proactive strategies, specifically in terms of maximum and average resource usage. This gives insight in choosing the best strategy when designing a system, considering constraints of available bandwidth and storage, specifically tailored for fast adaptation of cached context data in mobile environments, where context data access patterns can quickly change due to mobility. We extensively compared ACDC against traditional caching techniques (FIFO, LFU, and LRU) to assess performance improvements and total overhead introduced by our approach.

IV. REFERENCES

[1] B. N. Schilit, N. Adams, R. Want, “Context-Aware Computing Applications”, Proceedings of the Workshop on Mobile Computing Systems and Applications (WMCSA), pp. 85-90, 1994.

[2] Q. Jones, S. A. Grandhi, “P3 Systems: Putting the place back into social networks”, IEEE Internet Computing, vol. 9, no. 5, pp. 38-46, Sept.-Oct. 2005.

[3] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, P. Steggles, “Towards a better understanding of context and context-awareness”, Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing, pp. 304-307, 1999.

[4] P. Bellavista, A. Corradi, M. Fanelli and L. Foschini, “A Survey of Context Data Distribution for Mobile Ubiquitous Systems,” ACM Computing Surveys, vol. 44, no. 4, article no. 24, pp. 1-45, Aug. 2012.

[5] M. Baldauf, S. Dustdar, and F. Rosenberg, “A Survey on Context-aware systems”, International Journal of Ad Hoc and Ubiquitous Computing, vol. 2, no. 4, pp. 263-277, Jun. 2007.

[6] G. Chen, and D. Kotz, “A Survey of Context-Aware Mobile Computing Research”, Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, 2000.

[7] J. Hightower, G. and Boriello, “A Survey and Taxonomy of Location Systems for Ubiquitous Computing”, IEEE Computer, vol. 34, no. 8, pp. 57-66, Aug. 2001.

[8] T. Hofer, W. Schwinger, M. Pichler, G. Leonhartsberger, J. Altmann, W. Retschitzegger, “Context-Awareness on Mobile Devices - the Hydrogen Approach”, Proceedings of the 36th Annual Hawaii International Conference on System Sciences, pp. 10-19, 2003.

[9] U. Lee, E. Magistretti, M. Gerla, P. Bellavista, P. Liò, K.-W. Lee, “Bioinspired multi-agent data harvesting in a proactive urban monitoring environment”, Elsevier Ad Hoc Networks, vol. 7, no. 4, pp. 725-741, 2009.

[10] H. Chen, T. Finin, and A. Joshi, "An Intelligent Broker for Context- Aware Systems", Proceedings of Ubicomp, 2003.

[11] P. Fahy, and S. Clarke, "CASS – Middleware for Mobile Context-Aware Applications", Proceedings of the Workshop on Context Awareness at MobiSys, 2004.

[12] G. Chen, M. Li, D. Kotz, "Data-centric middleware for context-aware pervasive computing", Elsevier Pervasive and Mobile Computing, vol. 4, no. 2, pp. 216-253, Apr. 2008.

[13] K. Cho, I. Hwang, S. Kang, B. Kim, J. Lee, S. Lee, S. Park, J. Song, Y. Rhee, "HiCon: a hierarchical context monitoring and composition framework for next-generation context-aware services", IEEE Network, vol. 22, no. 4, pp. 34-42, Jul.-Aug. 2008.

[14] A. Derhab and N. Badache, "Data replication protocols for mobile adhoc networks: a survey and taxonomy", IEEE Communications Surveys & Tutorials, vol. 11, no. 2, pp. 35-51, Jun. 2009.

[15] P. Padmanabhan, L. Gruenwald, A. Vallur, and M. Atiquzzaman, "A survey of data replication techniques for mobile ad hoc network databases", The VLDB Journal, vol. 17, no. 5, pp. 1143-1164, Aug. 2008.

[16] A. Shaheen, and L. Gruenwald, "Group based replication for mobile ad hoc databases (GBRMAD)", Technical Report, University of Oklahoma, available at http:/ / www.cs.ou.edu/ ~database/ documents/ shaheen.pdf. Sept. 2010.

