Related Works

Mark Madler

1 KVS over RDMA

1.1 Kite: efficient and available release consistency for the datacenter

This is the real entry paper. This is a replicated KVS over RDMA with **Release Consistency**.[8]

1.2 FaRM: Fast Remote Memory

Super similar to the entry paper in that is is a KVS for RDMA, but this one is I beleive either disaggregated or not. Farm could really be classified as a KVS or even a protocol. This design always replicates state info. Provides **strict serializability**.[3]

1.3 FaRMv2: Fast General Distributed Transactions with Opacity

Just like FaRM but with opacity. Also providing **strict serializability**.[17]

1.4 HERD: Using RDMA efficiently for key-value services

Herd.[13]

1.5 Sherman

This paper [21] is disaggregated. Node-granularity. "consistent".

1.6 Pilaf

Linearizable data store. Self-correcting

1.7 scythe

KVS again.

1.8 HCL

Strictly serializabile KVS

1.9 Rolex

I don't think this one really works.

2 DSM systems

2.1 Scaling out NUMA-Aware Applications with RDMA-Based Distributed Shared Memory: MAGI

Page-based DSM. [10]

2.2 Efficient Distributed Memory Management with RDMA and Caching

cache-line granularity DSM.[2]

2.3 Distributed Shared Object Memory

object based granularity, release consistency... too old for RDMA[9]

2.4 Gengar: An RDMA-based Distributed Hybrid Memory Pool

This is object based dsm over rdma but with non-volatile memory as well using Intel Optane. Seems to also use this lease assignment idea like in [5] but is not page based.[4]

2.5 TreadMarks: shared memory computing on networks of workstations

Was implemented over IP, lazy release consistency I think. Not sure of granularity yet.[1]

2.6 LITE Kernel RDMA Support for Datacenter Applications

This is page based DSM using the kernel. [19]

2.7 MENPS: A Decentralized Distributed Shared Memory Exploiting RDMA

- Page based DSM
- Special Diff merging and page sharing
- Combine write notices and logical leases (what is that?)[5]

2.8 Argo DSM

Page-based DSM again but directory coherence. This was maybe the first RDMA-based DSM paper, at least thats what the authors allude to.[15]

2.9 GiantVM: A Novel Distributed Hypervisor for Resource Aggregation with DSM-aware Optimizations

Page-based DSM again but also works over TCP and RDMA[12]

2.10 Scalable RDMA performance in PGAS languages

This paper is for PGAS languages. Has an address hash table similar to LOCO for remote lookups.[6]

2.11 Misc PGAS languages probably

3 Protocols over RDMA for Consistency

3.1 Notes on PGAS and "protocols"

It seems like there are not agreed upon semantics on what is a protocol. MPI seems like a protocol but is it? PGAS is a memory model.

3.2 Odyssey: The Impact of Modern Hardware on Strongly-Consistent Replication Protocols

This paper is a summary of protocols used for RDMA communication. These protocols were used to enforce consistency and were tested with a series of KVSs. This paper is related to Kite (same authors) and Kite is one of the KVSs tested.[7]

3.3 Hermes: A Fast, Fault-Tolerant and Linearizable Replication Protocol

This paper [14] is one of the Protocols tested by the above paper Odyssey [7]. This protocol guarantees linearizablity and is designed to work on replicated store systems.

3.4 Hamband: RDMA Replicated Data Types

This paper [11] designed new RDMA data types that are replicated across nodes. This paper is sort of a protocol paper as it implements this protocol to keep replicated data through either relaxed or 'strong consistency'.

3.5 Evaluation of RDMA opportunities in an Object-Oriented DSM

Interesting result is that it proves that invalidation protocols are better suited for distributed systems. [20]

4 table i found

TABLE 1
Categories of RDMA-Based Storage Systems and Software Techniques

System Types	Related Works
Key-value Store	HERD [6] ccKVS [7] FaSST [8] Pilaf [9] RFP [10] HydraDB [11] C-Hint [12] DrTM [13] FaRM [14] Nessie [15] RStore [16] ScaleTX [17] Cell [18] Catfish [19] NAM-Tree [20] NVDS [21] FlatStore [22] RDMP-KV [23] RACE [24] RAMCloud [25] Sherman [32]
File System	CephFS [33] GlusterFS [34] Crail [35] NVFS [36] Octopus [5] Orion [37] FileMR [38] Assise [39] DeltaFS [40] GekkoFS [41] DAOS [42] PolarFS [43] Lustre [44] GPFS [45] BeeGFS [46] PVFS2 [47]
Distributed Memory	FaRM [14] RackOut [48] Grappa [49] InfiniSwap [50] Hotpot [51] Clover [52] AsymNVM [53] Kona [54] CoRM [55]
Databases	NAM-DB [56], [57] Chiller [58] PolarDB Serverless [59] D-RDMA [60] Zamanian *et al.* [61] Li *et al.* [62] HyPer [63] Barthels *et al.* [64] I-Store [65] L5 [66] Liu *et al.* [67]
Smart NICs	FlexNIC [68] KV-Direct [69] Lynx [70] StRoM [71] LineFS [72] Xenic [73] 1RMA [28] D-RDMA [60] HyperLoop [74]
Core Modules	**Related Works**
Communication Mode	DrTM-H [75] Cell [18] Catfish [19] Storm [76] DaRPC [77] HERD [6] FaSST [8] RF-RPC [78] ScaleRPC [17] Storm [76] Octopus [5] FlatStore [22] LITE [79] eRPC [80] Accelio [81] Mercury [82] X-RDMA [29] FLOCK [83] DFI [84] HatRPC [85]
Concurrency Control	DrTM [13] FaRM [14] Cell [18] NAM-Tree [20] Pilaf [9] RACE [24]
Fault Tolerance	HydraDB [11] Mojim [86] Orion [37] Tailwind [87] HyperLoop [74] DARE [88] APUS [89] Derecho [90] Odyssey [91] INEC [92] Aguilera *et al.* [93] Zamanian *et al.* [61]
Caching Resource Management	GAM [94] Aguilera *et al.* [95] DrTM [13] HydraDB [11] C-Hint [12] XStore [96] RACE [24] Kumar *et al.* [97] HERD [6] FaSST [8] FaRM [14] LITE [79] ScaleRPC [17] X-RDMA [29] FLOCK [83]

[16]

5 Loosely Related but Evaluated

5.1 CoRM: Compactable Remote Memory over RDMA

page based I think (re-read this)[18]

5.2 Rcmp: Reconstructing RDMA-Based Memory Disaggregation via CXL

page based and uses CXL, not comparable[22]

References

[1] Amza, C., Cox, A., Dwarkadas, S., Keleher, P., Lu, H., Rajamony, R., Yu, W., and Zwaenepoel, W. Treadmarks: shared memory computing on networks of workstations. *Computer 29*, 2 (1996), 18–28.

[2] Cai, Q., Guo, W., Zhang, H., Agrawal, D., Chen, G., Ooi, B. C., Tan, K.-L., Teo, Y. M., and Wang, S. Efficient distributed memory management with rdma and caching. *Proc. VLDB Endow. 11*, 11 (July 2018), 1604–1617.

[3] Dragojević, A., Narayanan, D., Hodson, O., and Castro, M. Farm: fast remote memory. In *Proceedings of the 11th USENIX Conference on Networked Systems Design and Implementation* (USA, 2014), NSDI'14, USENIX Association, p. 401–414.

[4] Duan, Z., Liu, H., Lu, H., Liao, X., Jin, H., Zhang, Y., and He, B. Gengar: An rdma-based distributed hybrid memory pool. In *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)* (2021), pp. 92–103.

[5] Endo, W., Sato, S., and Taura, K. Menps: A decentralized distributed shared memory exploiting rdma. In *2020 IEEE/ACM Fourth Annual Workshop on Emerging Parallel and Distributed Runtime Systems and Middleware (IPDRM)* (2020), pp. 9–16.

[6] Farreras, M., Almasi, G., Cascaval, C., and Cortes, T. Scalable rdma performance in pgas languages. pp. 1–12.

[7] Gavrielatos, V., Katsarakis, A., and Nagarajan, V. Odyssey: the impact of modern hardware on strongly-consistent replication protocols. In *Proceedings of the Sixteenth European Conference on Computer Systems* (New York, NY, USA, 2021), EuroSys '21, Association for Computing Machinery, p. 245–260.

[8] Gavrielatos, V., Katsarakis, A., Nagarajan, V., Grot, B., and Joshi, A. Kite: efficient and available release consistency for the datacenter. In *Proceedings of the 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming* (New York, NY, USA, 2020), PPoPP '20, Association for Computing Machinery, p. 1–16.

[9] Guedes, P., and Castro, M. Distributed shared object memory. In *Proceedings of IEEE 4th Workshop on Workstation Operating Systems. WWOS-III* (1993), pp. 142–149.

[10] Hong, Y., Zheng, Y., Yang, F., Zang, B.-Y., Guan, H.-B., and Chen, H.-B. Scaling out numa-aware applications with rdma-based distributed shared memory. *Journal of Computer Science and Technology 34* (2019), 94–112.

[11] Houshmand, F., Saberlatibari, J., and Lesani, M. Hamband: Rdma replicated data types. In *Proceedings of the 43rd ACM SIGPLAN International Conference on Programming Language Design and Implementation* (New York, NY, USA, 2022), PLDI 2022, Association for Computing Machinery, p. 348–363.

[12] Jia, X., Zhang, J., Yu, B., Qian, X., Qi, Z., and Guan, H. Giantvm: A novel distributed hypervisor for resource aggregation with dsm-aware optimizations. *ACM Trans. Archit. Code Optim. 19*, 2 (Mar. 2022).

[13] Kalia, A., Kaminsky, M., and Andersen, D. G. Using rdma efficiently for key-value services. In *Proceedings of the 2014 ACM Conference on SIGCOMM* (New York, NY, USA, 2014), SIGCOMM '14, Association for Computing Machinery, p. 295–306.

[14] Katsarakis, A., Gavrielatos, V., Katebzadeh, M. S., Joshi, A., Dragojevic, A., Grot, B., and Nagarajan, V. Hermes: A fast, fault-tolerant and linearizable replication protocol. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems* (New York, NY, USA, 2020), ASPLOS '20, Association for Computing Machinery, p. 201–217.

[15] Kaxiras, S., Klaftenegger, D., Norgren, M., Ros, A., and Sagonas, K. Turning centralized coherence and distributed critical-section execution on their head: A new approach for scalable distributed shared memory. In *Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing* (New York, NY, USA, 2015), HPDC '15, Association for Computing Machinery, p. 3–14.

[16] Ma, S., Ma, T., Chen, K., and Wu, Y. A survey of storage systems in the rdma era. *IEEE Transactions on Parallel and Distributed Systems 33*, 12 (2022), 4395–4409.

[17] Shamis, A., Renzelmann, M., Novakovic, S., Chatzopoulos, G., Dragojević, A., Narayanan, D., and Castro, M. Fast general distributed transactions with opacity. In *Proceedings of the 2019 International Conference on Management of Data* (New York, NY, USA, 2019), SIGMOD '19, Association for Computing Machinery, p. 433–448.

[18] Taranov, K., Di Girolamo, S., and Hoefler, T. Corm: Compactable remote memory over rdma. In *Proceedings of the 2021 International*

Conference on Management of Data (New York, NY, USA, 2021), SIGMOD '21, Association for Computing Machinery, p. 1811–1824.

[19] Tsai, S.-Y., and Zhang, Y. Lite kernel rdma support for datacenter applications. In *Proceedings of the 26th Symposium on Operating Systems Principles* (New York, NY, USA, 2017), SOSP '17, Association for Computing Machinery, p. 306–324.

[20] Veldema, R., and Philippsen, M. Evaluation of rdma opportunities in an object-oriented dsm. pp. 217–231.

[21] Wang, Q., Lu, Y., and Shu, J. Sherman: A write-optimized distributed b+tree index on disaggregated memory. In *Proceedings of the 2022 International Conference on Management of Data* (New York, NY, USA, 2022), SIGMOD '22, Association for Computing Machinery, p. 1033–1048.

[22] Wang, Z., Guo, Y., Lu, K., Wan, J., Wang, D., Yao, T., and Wu, H. Rcmp: Reconstructing rdma-based memory disaggregation via cxl. *ACM Trans. Archit. Code Optim. 21*, 1 (Jan. 2024).