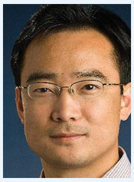
Processing Distributed Internet of Things Data in Clouds



Lizhe Wang
Chinese Academy of Sciences and China University of Geosciences

Rajiv Ranjan
Commonwealth Scientific and Industrial Research Organization

Recent studies by Cisco and IBM show that we generate 2.5 quintillion bytes of data per day, and this is set to explode to 40 yottabytes by 2020—that's 5,200 gigabytes for every person on earth.[1,2] Much of this data is and will be generated from Internet of Things (IoT) devices and sensors. IoT comprises billions of Internet-connected devices (ICDs) or "things," each of which can sense, communicate, compute, and potentially actuate, and can have intelligence, multimodal interfaces, physical/virtual identities, and attributes. ICDs can be sensors, RFIDs, social media, clickstreams, business transactions, actuators (such as machines/equipment fitted with sensors and deployed for mining, oil exploration, or manufacturing operations), lab instruments (such as a high energy physics synchrotron), and smart consumer appliances (TV, phone, and so on).

The IoT vision is to allow things to be connected anytime, anywhere, with anything and anyone, ideally using any path, network, and service. This vision has recently given rise to the notion of IoT big data applications that are capable of producing billions of datastreams and tens of years of historical data to provide the knowledge required to support timely decision making. These applications need to process and manage streaming and multidimensional data from geographically distributed data sources that can be available in different formats, present in different locations, and reliable at different levels of confidence.

IoT Big Data Application Requirements

The current generation of IoT big data applications (such as smart supply chain management, syndromic surveillance, and smart energy grids) combines multiple independent data analytics models, historical data repositories, and real-time datastreams that are likely to be available across geographically distributed datacenters (both private and public). For example, in a smart supply chain management IoT application, advanced analytics provides the next frontier of supply chain innovation. However, data management in supply chains is challenging because:

- datasets span multiple continents and are independently managed by hundreds of suppliers and distributors;
- datasets are updated in real time based on feeds from sensors attached to manufacturing devices and delivery vehicles; and
- customers express their sentiments regarding products via a mix of venues such as social media, product review portals, and blogs.

Companies must combine and analyze this distributed data along with contextual factors such as weather forecasts and pricing positions to establish which factors strongly influence the demand of particular products and then quickly take action to adapt to competitive and evolving environments. Similarly, syndromic surveillance IoT applications require churning through massive amounts of heterogeneous, real-time information available from social media, emergency rooms, health departments, hospitals, and ambulatory care sites to detect outbreaks of deadly diseases such as SARS, avian flu, cholera, and dengue fever.

Clearly, these IoT applications produce big datasets that can't be transferred over the Internet to be processed by a centralized public or private datacenter. The main reasons for this state of affairs are:

- the datasets have strict privacy, security, and regulatory constraints that prohibit their transfer outside the parent domain;
- the datasets flow at a volume and velocity too large and too fast to be processed by a single centralized datacenter as it could lead to high network communication overhead; and
- the analytics models and intelligence required to process the datasets are available across geographically distributed locations.

Despite the requirements posed by IoT big data applications, the capability of existing big data processing technologies and datacenter computing infrastructure is limited. For example, they can only process data on compute and storage resources within a centralized local area network, such as a single cluster within a datacenter. In addition, they don't provide mechanisms to seamlessly integrate data spread across multiple distributed heterogeneous data sources.

> The IoT vision is to allow things to be connected anytime, anywhere, with anything and anyone, ideally using any path, network, and service.

Finally, they can't ensure security and privacy-preserving processing of heterogeneous data governed by heterogeneous policies and access control rules.

State of the Art in Distributed IoT Data Processing

Existing big data processing technologies and datacenter infrastructures have varied capabilities with respect to meeting the distributed IoT data processing challenges.

Datacenter Cloud Computing Infrastructure Service Stack

Commercial and public datacenters such as Amazon Web Services and Microsoft Azure provide computing, storage, and software resources as cloud services, which are enabled by virtualized software/middleware stacks. Examples include virtual machine management systems such as Eucalyptus and Amazon Elastic Compute Cloud (EC2); image management tools such as the FutureGrid image repository[3]; massive data storage/file systems such as Google File System (GFS), the Hadoop distributed file system (HDFS), and Amazon Simple Storage Service (S3); and data-intensive execution frameworks such as Amazon Elastic MapReduce. In addition, FutureGrid (http://FutureGrid.org) and OpenStack provide software stack definitions for cloud datacenters.

On the other hand, private datacenters typically build basic infrastructure services by combining available software tools and services. This software includes cluster management systems such as Torque, Oscar, and Simple Linux Utility for Resource Management (Slurm); parallel file/storage systems such as storage area network/network-attached storage (SAN/NAS)[4] and Lustre (http://wiki.lustre.org); as well as data management systems such as the Berkeley Storage Manager (BeSTMan, https://sdm.lbl.gov/bestman) and dCache (www.dcache.org). In addition, some private datacenters are enabled for resource sharing with grid computing middleware, such as Globus toolkits, Uniform Interface to Computing Resources (Unicore, www.unicore.eu), and Lightweight Middleware for Grid Computing (gLite).

Massive Data Processing Models and Framework

The MapReduce paradigm has been widely used for large-scale data-intensive computing within datacenters due to its low cost, massive data parallelism, and fault-tolerant processing. The most popular implementation, Hadoop framework allows applications to run on large clusters and provides transparent reliability and data transfer. Other implementations include Compute Unified Device Architecture (CUDA),[5] field programmable gate array (FPGA),[6] virtual machines,[7] as well as streaming runtime,[8] grid,[8] and opportunistic environment.[9] Apache Hadoop on Demand (HOD) provides virtual Hadoop clusters over a large physical cluster based on Torque. MyHadoop provides on-demand Hadoop instances on high-performance computing (HPC) resources via traditional schedulers.[10] Other MapReduce-like projects include Twister (www.iterativemapreduce.org), Sector/Spear (http://sector.sourceforge.net), and All-pairs.[11]

Data Management Service across Datacenters

The following four storage service abstractions supported by cloud providers differ in how they store, index, and execute queries:

- Binary Large Object (Blob) for unstructured data such as Amazon S3 and Azure Blob;
- key-value storage such as HBase, MongoDB, and BigTable;
- message queuing systems such as SQS and Apache Kafka; and
- relational database management systems such as Oracle and MySQL, which support ACID (atomicity, consistency, isolation, durability) transactional properties.

Accordingly, several research efforts have integrated different cloud data storage services by providing a transparent interface. Examples are Simple Cloud API (http://simplecloud.com), Simple API for Grid Applications (SAGA) with an SRM interface, and some uniform services such as PDC@KTH's proxy service[12] and Open Grid Services Architecture Data Access and Integration (OGSA-DAI, www.ogsadai.org.uk) Web services. In addition, a number of third-party providers (DropBox, Mozy, and so on) simplify online cloud storage access.

Data-Intensive Workflow Computing

Typical data-intensive scientific workflow frameworks include Pegasus, Kepler, Taverna, Triana, Swift, and Trident. Various business workflow technologies have also been applied to data-intensive workflow systems. Examples include service orchestration with Business Process Execution Language (BPEL) and YAWL (Yet Another Workflow Language),[13] service choreography with Web Services Choreography Description Language (WS-CDL, www.w3.org/TR/ws-cdl-10), and service-oriented architectures.[14]

Benchmark, Application Kernels, Standards, and Recommendations

Several benchmarks and application kernels have been developed, including Graph 500 (www.graph500.org), Hadoop Sort (http://wiki.apache.org/hadoop/Sort) and Sort benchmark (http://sortbenchmark.org), MalStone,[15] Yahoo Cloud Serving Benchmark (http://research.yahoo.com/Web_Information_Management/YCSB), Google cluster workload (http://code.google.com/p/googleclusterdata), TPC-H benchmarks (www.tpc.org/tpch), BigDataBench, BigBench, Hibench, and PigMix, fueled by the need to analyze the performance of different big data technologies. These benchmark suites model workloads for stress testing one or more categories of big data processing technologies. Among these frameworks, BigDataBench is most comprehensive because it constitutes workload models for NoSQL, database management systems (DBMSs), SPEs (Stream Processing Engines), and batch processing frameworks. Primarily, BigDataBench targets the search engine, social network, and e-commerce application domains.

However, there are limited benchmarks and application kernels for heterogeneous datacenters. In fact, there's no agreement on available performance benchmarking for executing large-scale IoT applications across distributed datacenters. Actually, the lack of intercenter benchmarks and standards should be the key research agenda for the future. Currently, the National Institute of Standards and Technology (NIST), Open Grid Forum (OGF), Distributed Management Task Force (DMTF) Cloud working group, Cloud Security Alliance, and Cloud Standards Customer Council are all working on cloud standards.

Research Issues

Big IoT data processing across multiple distributed datacenters remains challenging, mainly because of technical issues related to basic service stacks for datacenter computing infrastructures, massive data processing models, trusted data management services, data-intensive workflow computing, and benchmarks.

Service Stacks in a Multidatacenter Computing Infrastructure

Despite significant advances, public cloud computing technologies are still technically challenging for serving large-scale IoT applications across datacenters. First, cloud technologies must be integrated into the resource management and file systems of existing private datacenter infrastructures to provision cloud services.

Massive Data Processing Models for Datacenters

There are several limitations in using MapReduce and Hadoop for large-scale distributed massive data processing. First, this framework is limited to compute infrastructures within a local area network or datacenter, and can't be directly used for large-scale IoT applications across geographically distributed datacenters. Second, MapReduce suffers from performance degradation due to the absence of a high-performance parallel and distributed file system that can seamlessly operate across multiple datacenters. Third, MapReduce uses a task "fork" mechanism that can't be directly deployed in traditional private datacenters with local task managers such as Torque and Globus, not to mention the lack of security models. Fourth, the limited semantics of MapReduce can't easily present the diverse parallel patterns of large-scale scientific applications. In addition, MapReduce and Hadoop aren't currently widely supported by data-intensive workflow systems, although there are some preliminary efforts.[16]

Optimized Data Management across Datacenters

Several research efforts have integrated heterogeneous types of cloud storage services by providing a transparent interface. However, these services and interfaces can't guarantee that data is secured both in motion and at rest, don't support automated ranking of competing storage services, and cannot handle uncertainties regarding cloud storage services and network routes.

To process and store massive datasets across geographically distributed storage services while providing required quality of service (QoS) guarantees raises several concerns.

First, current cloud storage services aren't secure by nature because of the inherent risk of data exposure, tempering, and denial of data access. Ensuring data confidentiality, integrity, and availability is a great concern.

Another issue concerns the intelligence to automate the choice of the best storage services and network routes for optimal application QoS. Existing quantitative criteria approaches applied optimization[17] and performance measurement techniques[18] for selecting cloud services. Other research focuses on static XML schema matching methods.[19]

The uncertainty of cloud storage services and network routes in a multiple datacenter environment is another major concern. Several reactive techniques rely on service state monitoring and action triggering to ensure QoS while adapting to run-time variation in resource loading and failures.[20,21] Some QoS prediction methods such as the Network Weather Service use both monitoring and forecasting. Another proposed network QoS-aware approach uses QoS profiling, modelling, and prediction.[22]

Data-Intensive Workflow Computing

IoT applications typically require distributed processing of data as a workflow that spans across multiple data processing services and repositories. Several open source products exist for running data-intensive workflows. However, current systems suffer from some limitations. For example, there's limited support for workflow walk across heterogeneous file systems, such as Lustre, HDFS, and GFarm. There's also limited support for MapReduce task and subworkflow in data-intensive workflows across distributed datacenters. Finally, data storage and management services aren't incorporated in the service-oriented framework for data-intensive workflow systems.

Benchmark and Application Kernels

Currently, there's no agreement on available performance for executing large-scale IoT applications in distributed datacenters. Even worse, there are currently no intercenter benchmark and application kernels or standards for running large-scale IoT applications on distributed datacenters.

Large-scale IoT applications need to process and manage massive datasets across geographically distributed datacenters. These applications need to be provisioned across multiple datacenters to exploit independent and geographically distributed data sources and IT infrastructure. The capability of existing data processing computing tools (for example, file systems, MapReduce, and workflow technologies), however, is optimized for single datacenter. Future research efforts will need to tackle the challenge of provisioning IoT applications across multiple datacenters by extending existing big data processing tools with the ability to process data across geographic locations; developing techniques for ensuring security and privacy of sensitive data; and developing intelligent techniques for application provisioning based on cost, performance, and other QoS requirements.

References

1. R. Pepper and J. Garrity, "The Internet of Everything: How the Network Unleashes the Benefits of Big Data," *Global Information Technology Report*, Cisco, 2014, pp. 35–42; http://blogs.cisco.com/wp-content/

uploads/GITR-2014-Cisco-Chapter.pdf.

2. “Bringing Big Data to the Enterprise,” IBM, http://www-01.ibm.com/software/in/data/bigdata.
3. J. Diaz et al., “FutureGrid Image Repository: A Generic Catalog and Storage System for Heterogeneous Virtual Machine Images,” *Proc. 3rd IEEE Int’l Conf. Cloud Computing Technology and Science* (CloudCom 11), 2011, pp. 560–564.
4. P. Radzikowski, “SAN vs DAS: A Cost Analysis of Storage in the Enterprise,” Capitalhead, 2008; http://capitalhead.com/articles/san-vs-das-a-cost-analysis-of-storage-in-the-enterprise.aspx.
5. B. He et al., “Mars: A MapReduce Framework on Graphics Processors,” *Proc. 17th Int’l Conf. Parallel Architectures and Compilation Techniques* (PACT 08), 2008, pp. 260–269.
6. Y. Shan et al., “FPMR: MapReduce Framework on FPGA,” *Proc. 18th Ann. ACM/SIGDA Int’l Symp. Field Programmable Gate Arrays* (FPGA 10), 2010, pp. 93–102.
7. S. Ibrahim et al., “Cloudlet: Towards MapReduce Implementation on Virtual Machines,” *Proc. 18th ACM Int’l Symp. High Performance Distributed Computing* (HPDC 09), 2009, pp. 65–66.
8. S. Pallickara, J. Ekanayake, and G. Fox, “Granules: A Lightweight, Streaming Runtime for Cloud Computing with Support for Map-Reduce,” *Proc. Cluster Computing and Workshops* (CLUSTER 09), 2009, pp. 1–10.
9. H. Lin et al., “Moon: MapReduce on Opportunistic Environments,” *Proc. 19th ACM Int’l Symp. High Performance Distributed Computing* (HPDC 10), 2010, pp. 95–106.
10. S. Krishnan, M. Tatineni, and C. Baru, *MyHadoop—Hadoop-on-Demand on Traditional HPC Resources*, tech. report, San Diego Supercomputer Center, Univ. of California, San Diego, 2011.
11. C. Moretti et al., “All-Pairs: An Abstraction for Data-Intensive Cloud Computing,” *Proc. IEEE Int’l Symp. Parallel and Distributed Processing* (IPDPS 08), 2008, pp. 1–11.
12. I. Livenson and E. Laure, “Towards Transparent Integration of Heterogeneous Cloud Storage Platforms,” *Proc. 4th Int’l Workshop Data-Intensive Distributed Computing* (DIDC 11), 2011, pp. 27–34.
13. C. Ouyang, M. Adams, and A.H.M. Hofstede, “Yet Another Workflow Language: Concepts, Tool Support, and Application,” *Handbook of Research on Business Process Modeling*, IGI Global, 2009, pp. 92–121.
14. K.K. Droegemeier et al., “Service-Oriented Environments for Dynamically Interacting with Mesoscale Weather,” *Computing in Science and Engineering*, vol. 7, no. 6, 2005, pp. 12–29.
15. C. Bennett, R. Grossman, and J. Seidman, *MalStone: A Benchmark for Data Intensive Computing*, tech. report TR-09-01, Open Cloud Consortium, 2009; http://malgen.googlecode.com/files/malstone-TR-09-01.pdf.
16. J. Wang, D. Crawl, and I. Altintas, “Kepler + Hadoop: A General Architecture Facilitating Data-Intensive Applications in Scientific Workflow Systems,” *Proc. 4th Workshop Workflows in Support of Large-Scale Science* (WORKS 09), 2009; doi:10.1145/1645164.1645176.
17. M. Hajjat et al., “Cloudward Bound: Planning for Beneficial Migration of Enterprise Applications to the Cloud,” *ACM SIGCOMM Computer Comm. Rev.*, vol. 40, no. 4, 2010, pp. 243–254.
18. A. Li et al., “CloudCMP: Comparing Public Cloud Providers,” *Proc. 10th Ann. Conf. Internet Measurement*, 2010, pp. 1–14.
19. A. Ruiz-Alvarez and M. Humphrey, “An Automated Approach to Cloud Storage Service Selection,” *Proc. 2nd Int’l Workshop Scientific Cloud Computing* (ScienceCloud 11), 2011, pp. 39–48.
20. L.M. Vaquero et al., “Dynamically Scaling Applications in the Cloud,” *SIGCOMM Computer Comm. Rev.*, vol. 41, no. 1, 2011, pp. 45–52.
21. Z. Shen et al., “CloudScale: Elastic Resource Sharing for Multi-Tenant Cloud Systems,” *Proc. 2011 ACM Symp. Cloud Computing*, 2011.
22. K. Alhamazani et al., “Cloud Monitoring for Optimizing the QoS of Hosted Applications,” *Proc. 4th IEEE Int’l Conf. Cloud Computing Technology and Science*, 2012, pp. 765–770.

LIZHE WANG *is a professor at the Institute for Remote Sensing and Digital Earth, Chinese Academy of Sciences, and at the School of Computer at the China University of Geosciences. Wang has a PhD in computer science from Karlsruhe University. He is a senior member of IEEE. Contact him at lizhewang@icloud.com.*

RAJIV RANJAN *is a senior research scientist, Julius Fellow, and project leader at the Commonwealth Scientific and Industrial Research Organization. At CSIRO, he leads research projects related to cloud computing, content delivery networks, and big data analytics for Internet of Things (IoT) and multimedia applications. Ranjan has a PhD in computer science and software engineering from the University of Melbourne. Contact him at rajiv.ranjan@csiro.au.*