BiobankCloud: Secure Storage, Sharing, and Processing of NGS Data

Alysson Bessani[5], Marc Bux[2], Joergen Brandt[2], Jim Dowling[1], Ali Gholami[1], Micheal Hummel[4], Mahmoud Ismail[1], Erwin Laure[1], Ulf Leser[2], Jan-Eric Litton[3], Roxanna Martinez[3], Salman Niazi[1]

[1] KTH - Royal Institute of Technology,
{jdowling, gholami, maism, erwinl, smkniazi}@kth.se
[2] Humboldt University
{leser, bux, joergen.brandt}@informatik.hu-berlin.de
[3] Karolinska Institute
{Jan-Eric.Litton, Roxanna.Martinez}@ki.se
[4] Charite
{Michael.Hummel}@charite.de
[5] University of Lisbon
{bessani}@di.fc.ul.pt

Abstract. Biobanks store and catalogue human biological material that is increasingly becoming digitized using technologies such as whole genome sequencing. There is, however, a growing Biobank bottleneck: a lack of open-source software systems that can securely scale to handle the coming wave of genomic data from Next-Generation Sequencing machines. In the BiobankCloud project, we are building an open-source Hadoop-based platform for the secure storage, sharing, and processing of genomic data. We have extended Hadoop to include support for multi-tenancy (studies), reduced data sizes with erasure coding, support for extensible and consistent metadata, a scalable scientific workflow framework, and support for securely sharing data across Hadoop instances. We demonstrate the capabilities of BiobankCloud in a Laboratory Information Managmenent System with support for 2-factor authentication and UI-support to all major services.

1 Introduction

Biobanks store and catalog human biological material from identifiable individuals for both clinical and research purposes. Recent advances in Next-Generation Sequencing (NGS) technology has meant that there is an increasing demand to sequence the human biological material stored in Biobanks. Both research projects and clinical health-care systems use Biobanks to store samples that are sequenced using techniques from genotyping to whole-genome sequencing (WGS). Biobanks' computer systems have traditionally managed only metadata associated with samples, such as pseudo-identifiers for patients, sample collection information, study infromation, and data concerning samples. Alongside

this metadata, we now increasingly need to store genomic data, which requires anything from MBs (genotyping) up to several hundred GBs (WGS) of data per sample. The data storage requirements for large-scale WGS sequencing projects, such as the 100,000 genomes project by Genomics England, now scale to many PBs, and are so large that researchers are looking at cost-effective Big Data solutions based on commodity hardware, such as Hadoop. When data volumes grow past several TBS, traditional database technology (including sharded relational databases) is no longer viable, as more data needs to be moved from storage to compute nodes that is possible with current networking technology. The solution provided by platforms such as Hadoop is to move computation to where the data is located, exploiting the principle of *data locality*. That is, jobs are parallelized across many nodes and each job loads its (large) input data primarily from disk subsystems, while network I/O is used transfer in subsequent processing and sorting steps on, typically, smaller data volumes relative to the input data size.

In this paper, we introduce BiobankCloud, an integrated platform, based on Hadoop, for the secure storage, processing, and sharing of genomic data and associated metadata. BiobankCloud is a platform-as-a-service that can be automatically deployed on public clouds, private clouds or bare-metal servers. As part of BiobankCloud, we also provide a Laboratory Information Management Service (LIMS) as software-as-a-service (SaaS). The LIMS has an integrated User Interface (UI) for authenicating/authorizing users, managing data, designing and searching for metadata, and support for running workflows and analysis jobs on Hadoop. The LIMS hides much of the complexity of the Hadoop backend, and supports multi-tenancy through first-class support for *Studies*, *SampleCollections* (DataSets), *Samples*, and *Users*.

In BiobankCloud, we have developed our own Hadoop distribution, Hadoop Open Platform-as-a-Service (Hops), with improved scalability and customizability properties, enabled by a new metadata storage system based on a distributed in-memory database.

Existing scientific workflow management systems are typically custom-built and have not been designed for data parallel processing with data locality. For performance reasons, their architectures and file formats are typically flat (monolithic). None of the main file formats in genomics (fastq, BAM/SAM, CRAM, and VCF) have support for data parallel processing because of their assumption of centralized metadata (e.g., in the file header). In BiobankCloud, we provide a scientific workflow management system, SaasFee, as a native YARN/Hadoop 2nd-level scheduler that provides a bridge between support for existing file formats and data parallel processing. SaasFee can both speedup workflows, such as a ??x speedup for NGS varient calling on XX machines, and scale-out to support larger clusters.

2 Related work

There are a couple of frameworks for data parallel processing of genomic data. Adam, Halvade, Seal, PigSeq, Spork?

Arvados is a scale-out platform designed to manage genomic and biomedical data. However, it is based on a custom storage service (Keep) and a custom computation service (Crunch), instead of an open platform such as Hadoop, and has GPLv3 licensing which will limit its use in commercial services.

Hadoop has support for Kerberos to secure services interally within Hadoop. For access control for external users, there are a number of different frameworks available, specialized for different services. Apache Ranger is a general attribute-based access control that could, in theory, be applied to all services. However, existing attribute-based access control solutions cannot scale to process the number of access control requests per second that would be generated by HDFS's NameNode and YARN's ResourceManager. Other access control servcices include Apache Sentry which provides role-based access control (RBAC) for databases and tables in Hive. Finally, Apache Knox provides RBAC for REST APIs that are proxies for Hadoop services, such as HDFS, YARN, MapReduce, and HBase. Apache Knox can also act as a gateway to more than one Hadoop cluster.

3 BiobankCLoud Regulatory Framework

4 LIMS

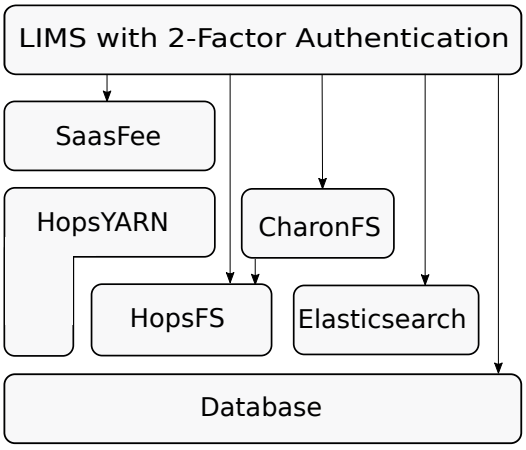

Fig. 1: BiobankCloud Architecture

In BiobankCloud, we are using our distribution of the Hadoop Filesystem (HDFS), HopsFS, to store large genomic files. HopsFS scales to store 100s of millions of files. There is a need to organize the files in such a manner that they can easily be logically grouped and searched. Typically, such functionality requires metadata for the files and directories, such as filename and last accessed

time. For biological samples, much more extensive metadata is required for files that relate to biological samples. We need information such as the sample collection it belongs to, the type of sample, and donor information. BiobankCloud enables Biobankers who are not programmers to design such metadata, link it to samples, and then edit sample metadata using UI support. The metadata is transparently indexed to enable free-text searching for samples, sample collections, and studies. Our solution is based on the industry-standard Elasticsearch platform. Importantly, we guarantee the metadata's integrity by implementing an eventually consistent replication model that asynchronously copies updates to metadata from our distributed database to Elasticsearch. The distributed database uses foreign keys to guarantee the integrity of the metadata and the referrent genomic data. Mutations to the metadata or removal of the sample will mutate or remove the metadata in Elasticsearch within seconds.

Biobankers can use a web application to design new tables that are transparently added to the database and have foreign-key constraints on existing files or directories in the system, thus maintaining the integrity of the metadata. Our system automatically exports metadata to Elasticsearch from where it can be searched using freetext by the user. We support both the automated indexing of files and directories in Elasticsearch as well as custom-designed metadata.

BiobankCloud introduces **DataSets** as a new abstraction to Hadoop, where a DataSet consists of a related group of directories, files, and extended metadata. DataSets can be indexed and searched and are the basic unit of data management in BiobankCloud; all user-generated files or directories belong to a single DataSet. In Biobanking, a sample collection would be a typical example of a DataSet.

To allow for access control of users to DataSets, which is not inherent in the DataSet concept, we introduce the notion of **Studies**. A Study is a grouping of researchers and DataSets with role-based access control where different researchers can be given different access rights to DataSets. The basic user roles we provide reflect the European Data Protection Directive, with a DataOwner (data controller) and a DataScientist (data processer). DataSets can be shared between Studies (when the necessary security, legal, and ethical conditions for sharing are in place). In BiobankCloud, we use the access control mechanism of HopsFS to implement the Study- and DataSet-based authorization model.

5 Security Model

The BioBankCloud environment deploys strong security features for concerns such as confidentiality, integrity and non-repudiation [1] of data access. This includes authentication, authorization, and auditing. The system allows defining different roles with different access privileges. In designing the system, we applied the Cloud Privacy Threat Modeling [2] approach to identify the privacy requirements of processing sensitive biomedical data.

Figure 2 shows the different components of the employed security mechanisms. All BioBankCloud services are protected behind the firewall and only accessible through the secure interfaces over HTTPS channels.

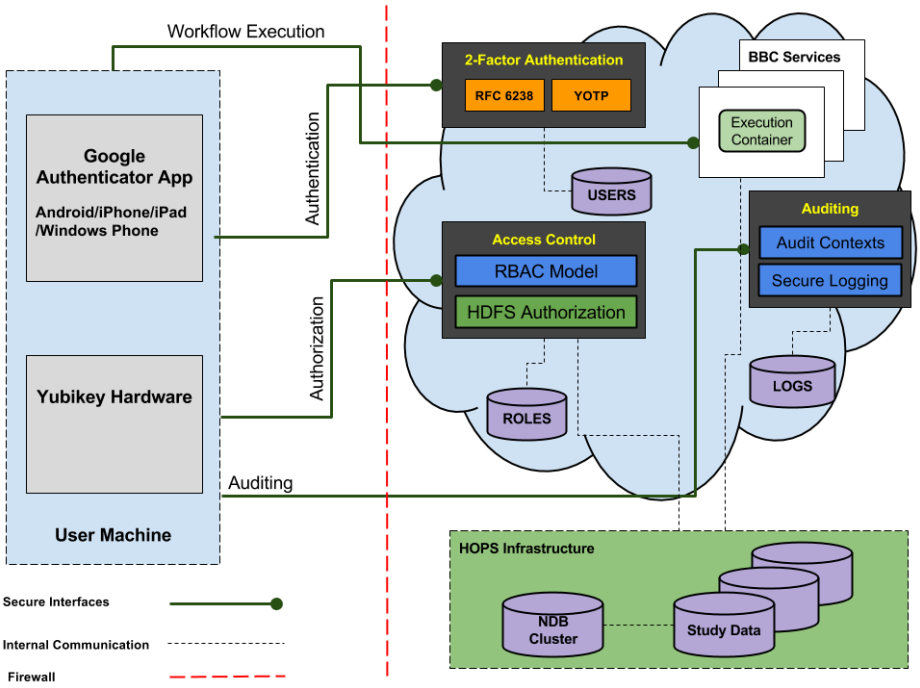

Fig. 2: Security Architecture of the BiobankCloud

5.1 2-Factor Authentication

The authentication services map the person accessing the platform to a user identity. We provide 2-factor authentication using smart mobile devices or Yubikey [6] hardware tokens to support different groups of users. Users send authentication requests via a Web browser to the authentication service that runs instances of the time-based one-time password (TOTP) and Yubikey one-time password (YOTP) protocols.

Mobile Authentication The mobile user supplies an one-time generated password by the Google Authenticator [7] in addition to a simple password that was

[6] Yubikey Manual,http://www.yubico.com
[7] Google Authenticator, https://code.google.com/p/google-authenticator/

decided during the account registration. The login page authenticates the user to to the platform using the TOTP module as an implementation of the RFC 6238 [8].

Yubikey Authentication The Yubikey enters the Yubikey device into a USB port. The user enters the simple password that was decied during the account registration and pushes the Yubikey button. The Yubikey login page authenticates the user to the platform via the YOTP module.

5.2 Access Control

The access control component ensures authorized access to genomics data as internal objects or different services within the platform. This is accomplished through a role-based access control (RBAC) model and authorization of data access on the HDFS, for example, a data owner adds/revokes members to a study and assigns privileges to access the study.

Role-Based Access Control The proposed RBAC model contains information about the roles of individuals within the organization and the associated levels of access to services.

- Administrator: group of users who acts as the platform manager and Ethics Board.
- Auditor: group of users with access to audit trails for auditing.
- Data Provider: group of users who create studies, upload data and assign members to studies.
- Guest: general visitors to the platform who are able to request an account to use the services.
- Researcher: users of the platform that can join a study to run workflows. Researchers also can become data providers through creating a new study and uploading data to the platform.

HDFS Supported Authorization [Ali: I made this part to be a subsubsection to fit better into the security figure. But it's also no issue to have it as a subsection.]

5.3 Auditing Service

Finally, the auditing service enables the platform administrator or an external auditor to discover the history of accessing the platform to detect any violation to a policy. It includes several contexts such as role, account, study, and login audits. The secure login service assures that actions that are taken by the users are registered for tracing and auditing purposes. Each log event contains informat ion such as initiator, target, IP/MAC addresses, timestamp, action, and outcome.

[8] Time-Based One-Time Password (TOTP), http://tools.ietf.org/html/rfc6238

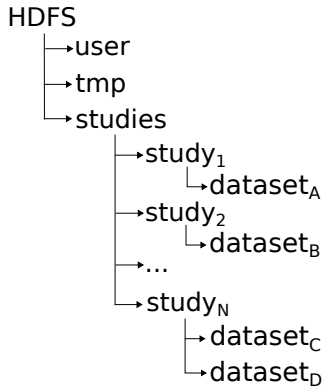

Fig. 3: Filesystem structure in HDFS containing Studies and DataSets.

6 Hadoop Open Platform-as-a-Service (Hops)

We have moved metadata from the heap of a Java Virtual Machine to an in-memory, no-shared-state, distributed database, called MySQL Cluster [3]. We ensure the consistency of the filesystem metadata by implementing serialized transactions on well-ordered operations on metadata [4]. We do not need Zookeeper, as we have developed a leader-election service based on the database [5].

7 SAASFEE

To process the vast amounts of genomic data stored in today's Biobanks, researchers have a diverse ecosystem of tools at their disposal [7]. Depending on the research question at hand, these tools are often used in conjunction with one another, resulting in complex and intertwined analysis pipelines. Scientific workflow management systems (SWfMSs) facilitate the design, refinement, execution, monitoring, sharing, and maintenance of such analysis pipelines. SAASFEE [8] is a SWfMS that supports the scalable execution of arbitrarily complex workflows. It encompasses the functional workflow language Cuneiform as well as Hi-WAY, a higher-level scheduler for Hadoop YARN.

Analysis pipelines for large scale genomic data employ many different software tools and libraries with diverse Application Programming Interfaces (APIs). At the same time the growing data sets to be analyzed necessitate parallel and distributed execution of these analysis pipelines. Thus, the methods for specifying such analysis pipelines need to meet both concerns, integration and parallelism equally. The functional workflow Language Cuneiform has been proposed to meet these requirements [6]. Cuneiform allows the integration of software tools and libraries with APIs in many different programming languages. This way, command-line tools (e.g. Bowtie) can be integrated with the same ease as R libraries (e.g. CummeRbund). By partitioning large data sets and processing these partitions in parallel, data parallelism can be expoited in addition to

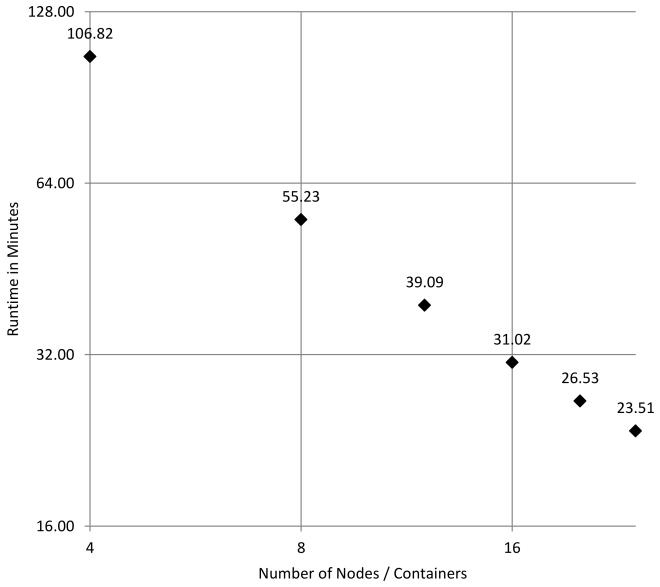

Fig. 4: Scalability experiment for the SAASFEE software stack. A variant calling workflow has been scaled out on up to 24 computers. Both axes, the runtime in minutes and the number of nodes are on a logarithmic scale (published in Brandt et al. 2015 [6]).

task parallelism to speed up computation. Cuneiform automatically detects and exploits data- and task-parallelism in a workflow specification. Editing and debugging workflows is supported by the tools and visualization features provided with the Cuneiform interpreter.

Hi-WAY is a higher-level scheduler for executing scientific workflows on Hadoop YARN. It provides a selection of established scheduling policies conducting task placement based on (a) the locality of a task's input data to diminish network load and (b) task runtime estimation based on past measurements to utilize resources efficiently. To enable repeatability of experiments, Hi-WAY generates exhaustive provenance traces during workflow execution, which can be shared and re-executed or archived in a database. One of the major distinctive features of SAASFEE is its strong emphasis on integration of external software. This is true for both Cuneiform, which is able to integrate foreign code and command-line tools, and Hi-WAY, which is capable of running not only Cuneiform workflows, but also workflows designed in the SWfMSs Pegasus [10] and Galaxy [11].

To verify the scaling behaviour of SAASFEE for large-scale use cases relevant for biobanks, we specified and ran a variant calling pipeline on 10 GB compressed whole genome sequencing reads from the 1000 genomes project. These reads were aligned against a reference workflow, variants were called, and the resulting sets of variants were annotated using publicly available databases. Figure 4 shows the

scaling behaviour of the resulting workflow. Within the limits of the setup chosen, linear scaling behaviour could be achieved for the variant calling workflow.

8 Sharing Data Between Clusters with CharonFS

CHARON is a cloud-backed file system capable of storing and sharing big data in a secure, reliable, and efficient way using multiple cloud providers and storage repositories. It is secure and reliable because it does not require trust on any single entity, and it supports the storage of different types of data in distinct locations to comply with required privacy premises. Two distinguishing features of CHARON are its serverless design (no client-managed server is required in the cloud) and its efficient management of large files (by employing prefetching, cache, and background writes). The complete description of CHARON architecture and its internal protocols is available in the Deliverable D4.2 of the BiobankCloud project [?].

Figure ?? illustrates a deployment scenario where two biobanks store their data in local repositories, in single public cloud providers, and in a resilient cloud-of-clouds. In this scenario, the namespace tree has six nodes: directories `d1` and `d2`, and files `A`, `B`, `C`, and `D`. The namespace is maintained in the cloud-of-clouds, together with file `B`. File `D`, less critical, is kept in a single cloud. File `A` is stored locally because it cannot leave Biobank 2. File `C` is shared between the two sites (e.g., in the same country), thus being stored in both of them.

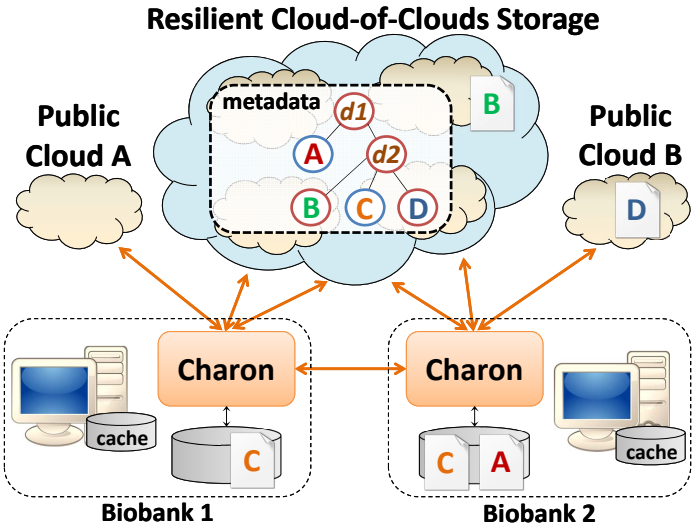

Fig. 5: CHARON overview.

The BiobankCloud platform will be able to process data available in the Hops-FS, which has a view of all data being stored in a single datacenter.

CHARON will perform the inter-datacenter tasks and the processes between biobanks and public clouds. In the next section, we discuss the integration between these two storage systems and evaluate the most appropriate integration scenario for the BiobankCloud platform.

9 Conclusions

In this paper, we introduced the BiobankCloud platform, a Hadoop-based platform that provides a number of features necessary for Biobanks to adopt Hadoop-based solutions for managing NGS data. Critical safety features that we introduced for managing sensitive data include multi-tenancy for the isolation of Study data and secure access through 2-factor authentication. Next-generation data management systems for NGS data must be massively scalable. We introduced our scalable storage service, HopsFS, our processing framework, Hops-YARN, and our framework for scalable bioinformatics workflows, Saasfee. We also showed the integration of explicity metadata design and management in the platform, ensuring the integrity of metadata and supporting free-text search using extended metadata. Finally, in CharonFS, we showed how we can leverage public clouds to share data securely between clusters.

10 Acknowledgements

This work funded by the EU FP7 project "Scalable, Secure Storage and Analysis of Biobank Data" under Grant Agreement no. 317871.

References

1. A. Gholami, J. Dowling, and E. Laure, "A security framework for population-scale genomics analysis," 2015. The International Conference on High Performance Computing and Simulation.
2. A. Gholami, A.-S. Lind, J. Reichel, J.-E. Litton, A. Edlund, and E. Laure, "Privacy threat modeling for emerging biobankclouds," *Procedia Computer Science*, vol. 37, no. 0, pp. 489 – 496, 2014. The 5th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2014)/ The 4th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2014)/ Affiliated Workshops.
3. M. Ronström and J. Oreland, "Recovery Principles of MySQL Cluster 5.1," in *Proc. of VLDB'05*, pp. 1108–1115, VLDB Endowment, 2005.
4. K. Hakimzadeh, H. Peiro Sajjad, and J. Dowling, "Scaling hdfs with a strongly consistent relational model for metadata," in *Distributed Applications and Interoperable Systems* (K. Magoutis and P. Pietzuch, eds.), Lecture Notes in Computer Science, pp. 38–51, Springer Berlin Heidelberg, 2014.
5. S. Niazi, M. Ismail, G. Berthou, and J. Dowling, "Leader election using newsql database systems," in *Distributed Applications and Interoperable Systems - 15th IFIP WG 6.1 International Conference, DAIS 2015, Held as Part of the 10th International Federated Conference on Distributed Computing Techniques, DisCoTec 2015, Grenoble, France, June 2-4, 2015, Proceedings*, pp. 158–172, 2015.

6. J. Brandt, M. Bux, and U. Leser, "Cuneiform: A functional language for large scale scientific data analysis," in *Proceedings of the Workshops of the EDBT/ICDT*, vol. 1330, (Brussels, Belgium), pp. 17–26, March 2015.
7. S. Pabinger, A. Dander, M. Fischer, R. Snajder, M. Sperk, M. Efremova, B. Krabichler, M. R. Speicher, J. Zschocke, and Z. Trajanoski, "A survey of tools for variant analysis of next-generation genome sequencing data," *Briefings in Bioinformatics*, vol. 15, pp. 256–278, Mar. 2014.
8. M. Bux, J. Brandt, C. Lipka, K. Hakimzadeh, J. Dowling, and U. Leser, "SAASFEE: Scalable Scientific Workflow Execution Engine," in *VLDB Demonstrations Track, forthcoming*, (Hawaii, USA), 2015.
9. J. Brandt, M. Bux, and U. Leser, "Cuneiform – A Functional Language for Large Scale Scientific Data Analysis," in *Workshop on Algorithms and Systems for MapReduce and Beyond, in conjunction with EDBT/ICDT Conference*, (Brussels, Belgium), 2015.
10. E. Deelman, K. Vahi, G. Juve, M. Rynge, S. Callaghan, P. J. Maechling, R. Mayani, W. Chen, R. F. da Silva, M. Livny, and K. Wenger, "Pegasus: A Workflow Management System for Science Automation," *Future Generation Computer Systems*, vol. 46, pp. 17–35, 2015.
11. J. Goecks, A. Nekrutenko, and J. Taylor, "Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences," *Genome Biology*, vol. 11, p. R86, Jan. 2010.