Cloud Computing

2

Bringing Computational Power to Medical Genetics

Lukas Forer, Sebastian Schönherr, Hansi Weißensteiner, Günther Specht, Florian Kronenberg, and Anita Kloss-Brandstätter

Abstract

Computer science plays a key role in today's genetic research. Next-generation sequencing technologies produce an enormous amount of data, pushing genetic laboratories to the limits of data storage and computational power. Therefore, new approaches are needed to eliminate these shortcomings and provide possibilities to use current algorithms in the area of bioinformatics with improved usability. A possible starting point is cloud computing with the opportunity to use linked computer systems and services on demand. Thus, huge amounts of data can be analysed much faster and more efficiently than by utilising a single computer system. This chapter gives the reader an overview about cloud computing, discusses its challenges and opportunities and shows existing solutions in the field of genetics to gather some hands-on experience.

2.1 Introduction

In recent years computer science became an essential part in the field of genetics. Especially through the advent of next-generation sequencing (NGS) technologies the amount of data is growing significantly, exceeding all known dimensions. For instance, to store the data of one complete human genome in raw format with 30 times coverage, ~30 TB (30,000 GB) of data is produced.[1] In the area of copy number variations (CNVs), a possible cause of many complex genetic disorders, high-throughput algorithms are needed to process and analyse several hundred gigabytes of raw input data, yielding to a calculation time of up to 1 week for a typical population study with thousands of subjects.

To emphasise that computer hardware can currently not keep pace with the progress in DNA sequencing, Fig. 2.1 shows a comparison between the trend in the reduction of DNA sequencing costs and the trend of Moore's law. Moore's law describes the development of computer processors

Lukas Forer, Sebastian Schönherr and Hansi Weißensteiner contributed equally to this work.

A. Kloss-Brandstätter (✉)
Division of Genetic Epidemiology, Department of Medical Genetics, Molecular and Clinical Pharmacology, Innsbruck Medical University, Schöpfstraße 41, 6020 Innsbruck, Austria
e-mail: anita.kloss@i-med.ac.at

[1] Using Illumina's 1G platform, including all image data.

Z. Trajanoski (ed.), *Computational Medicine*,
DOI 10.1007/978-3-7091-0947-2_2,

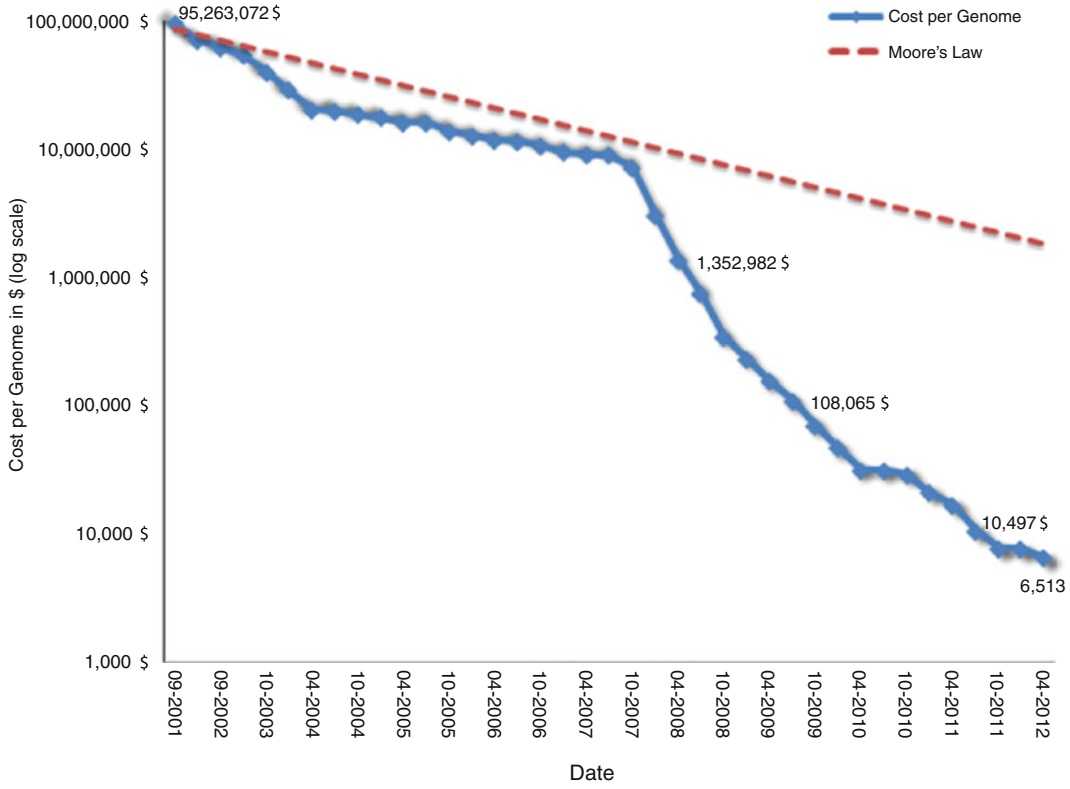

Fig. 2.1 Falling costs of DNA sequencing (*blue*) compared with Moore's law (*red*). Moore's law is only used as a reference value here. Data for sequencing costs are derived from Wetterstrand (2011)

and predicts the improvement of computer hardware. Compared to Moore's law, DNA sequencing costs are falling even faster which clearly yields to a flood of new DNA data. Thus, tasks like the execution of an algorithm, the storage of datasets on a local computer or processing data files in spreadsheet applications like Microsoft Excel or Open Office are not feasible anymore. This remarkable increase of data and time causes genetic departments to consider new ways of importing and storing their data as well as to improve the performance of current algorithms.

Computer clusters in the form of linked computers consisting of several hundred processors and huge amount of memory capacities have the potential to solve these issues. Unfortunately, small- to medium-sized genetic research institutes can often hardly afford the acquirement and maintenance of own computer clusters. Compared to the amount of analysis tasks these institutes have to accomplish per year, buying and maintaining their own computer would exceed the budget.

Using *clusters on demand* or with other words using a *public cloud approach* provides a good opportunity to circle these issues. The user is capable to rent as many computer nodes as needed from a cloud vendor to store data or to solve a computational problem. For instance, to solve a simple statistical analysis one fast machine in the cloud would be enough. Otherwise, for a complex alignment task the user would need several nodes at the same time to solve a problem efficiently. Almost all cloud vendors are based on a *pay-per-use* model in which the user pays for the time computer nodes are up and running. Moreover, the physical infrastructure is completely hidden from the end user, maintained and secured by professionals.

In the last years several bioinformatics projects and algorithms were developed, aimed at solving problems using current cluster architectures and

paradigms. For future developments it is anticipated that the trend to use cluster architectures for storing and analysing data will continuously increase.

This chapter should give the reader a better understanding of cloud computing in general and describes its application in the field of genetics, especially using cluster infrastructure on demand.

2.2 What Is Cloud Computing?

Cloud computing became a buzzword in recent years with the promise to solve major problems of today's information technology. The basic idea behind cloud computing is nothing new since cloud services are already components of a daily routine: checking emails, searching information on the World Wide Web or browsing in social networks are typical cloud functions. The storage and the processing of data itself is taking place somewhere in a remote data centre or with other words in a cloud. The cloud or cloud computing, originally a metaphor for the Internet, describes nowadays basically an abstraction of the underlying infrastructure it represents (Rittinghouse and Ransome 2009). The end user is able to access it without concerning about technical details and administrative issues. In a nut shell, cloud computing offers the possibility to provide IT infrastructures, i.e. computer hardware and software, dynamically to end users.

2.2.1 A Short History

In the end of the 1950s computer time-sharing technology was promoted as the future. This technology made it possible to share a single computing resource among many users at the same time. Thus, a more efficient usage of the resources could be achieved and the overall execution time was minimised.

In the mid 1960s, the time-sharing concept was evaluated and new architectures based on virtual machines ("simulated machines") were proposed, pushed especially by IBM.[2] Several virtual machines were able to run simultaneously on the same underlying physical machine, sharing common resources like main memory or the processor unit (CPU) in a fair way. Especially through technologies like XEN[3] or VMWare,[4] virtualization made its way to computer systems and even to normal desktop computers. Virtualization can be seen as the base for cloud computing by dividing physical server systems into as many virtual systems as reasonable and providing them to end users.

2.2.2 Public, Private and Hybrid Clouds

When talking about clouds nowadays, it is important to distinguish between *public* and *private clouds*: public clouds are accessible via the Internet, and private clouds are using own computer infrastructure that is not publicly available. Moreover, both approaches can be combined to a *hybrid cloud*, where the cloud is located inside a company with the possibility to replicate data on a public cloud.

2.3 Cloud Types

Cloud computing affects all levels of today's IT. When trying to classify cloud computing, three major categories can be defined: The *software cloud* (Software as a Service, SaaS) covers applications like access to search engines, email services or social networks. As a second category the *platform cloud* (Platform as a Service, PaaS) involves web services, backup possibilities or frameworks to develop and to share web applications (e.g. Google App Engine[5] or Microsoft Azure[6]). The *infrastructure cloud* (Infrastructure as a Service, IaaS) provides access to servers and storage possibilities

[2] IBM: http://www.ibm.com.

[3] XEN: http://www.xen.org.

[4] VMWare: http://www.vmware.com.

[5] Google App Engine: https://developers.google.com/appengine/.

[6] Microsoft Azure: http://www.windowsazure.com.

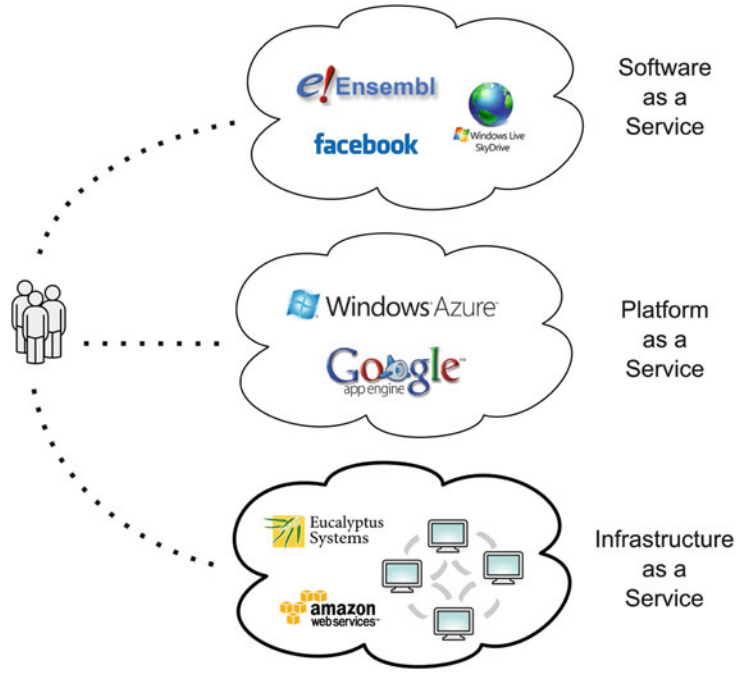

Fig. 2.2 Three categories of cloud computing: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS)

(see Fig. 2.2). This chapter refers mainly to the infrastructure cloud and its possibilities to gain access to computational power for companies as well as for scientific research institutes.

2.3.1 Software as a Service

SaaS delivers software solutions via the Internet thus eliminating the need to install or run an application on the local computer. When using SaaS data is permanently stored on remote computer systems, applications are delivered for example through a browser and the data is cached temporarily on client side. As the data is stored in the cloud and not on a local computer, the user is able to access the software from everywhere with access to the Internet. Beside the common use cases discussed in the introduction, examples for SaaS in the area of bioinformatics are the *Ensemble Genome Browser*[7] or the *UCSC Genome Browser*.[8]

[7] Ensemble Genome Browser: http://www.ensembl.org.
[8] UCSC Genome Browser: http://genome.ucsc.edu.

2.3.2 Platform as a Service

The PaaS approach delivers a platform with all facilities provided to build and deliver web applications. The application is able to utilise infrastructure from vendors automatically. The amount of resources used depends on how many users are currently using the application. For this purpose the platform analyses automatically its workload and activates or deactivates resources. An example of a PaaS architecture is *Google's App Engine* which provides the possibility to develop and host web applications without the need to buy servers. Web hosting, backup possibilities and database services can be seen as further typical platform cloud functions.

2.3.3 Infrastructure as a Service

IaaS describes the possibility to rent computer hardware ("infrastructure") from different vendors like *Amazon*, *Rackspace*, *Terremark* and many more. These instances can be accessed and completely controlled by the end user.

Instances in the infrastructure cloud are mainly virtualized systems.[9] As mentioned earlier, they are called virtualized since one physical instance shares its resources like CPU or main memory among several virtual instances, isolated and encapsulated from each other. This clearly yields to cost savings in hardware, maintenance and administration. The user has the possibility to acquire several instances at once and define a *cluster architecture* among the instances to conduct computational intensive tasks. A cluster describes a linkage of several computer instances, in which a specific work is distributed. Several issues like data security, scalability, data transfer and usability of current systems have to be taken into consideration when using IaaS. These issues are discussed in detail in the next section.

2.4 Challenges and Opportunities

As most techniques, also cloud approaches involve challenges and opportunities. The end user should be aware that it does not always make sense to work within a public cloud environment. Not every application can be run in the cloud efficiently, and before data is stored and processed in the cloud the following advantages and disadvantages of a cloud-based approach should be weighted.

2.4.1 Security

One of the big challenges for information technology is to secure public clouds in order to increase their trustability. As a survey showed, missing security is *the* knockout criterion for most users (Rittinghouse and Ransome 2009). Thus, it has to be assured that data is stored securely and that only entitled users are able to access it. The more confidential data gets, the more important data security becomes. For example, data is often used in unpublished research results and a lot of users do not feel comfortable putting such data on third party machines. Furthermore, if data is subject to regulations, special security rules and safeguards have to be guaranteed (Markovich 2010).

On the other side, a cloud-based approach could also provide opportunities for researchers. Data in a public cloud can be provided for everyone with the opportunity to share and reproduce results. Compared to a local computer a cloud approach is often more secure, since cloud instances are maintained and physically secured by professionals (Holland 2011). In 2008 two million laptops were stolen in the USA, not to mention mislaid USB keys with private company data. Often these stolen laptops contain highly sensitive data. In June 2011 a computer of the National Health Service in Liverpool was lost containing highly sensitive and unencrypted details relating to 8.63 million individuals (Clark 2011). On cloud systems, security updates are installed by vendors and the data is stored in a redundant manner. Furthermore, encryption technologies are fundamental for the success of cloud computing. Only an encrypted communication guarantees that transmitted data cannot be eavesdropped. There are a lot of strategies how to fulfil this issue and software solutions like the open source authentication and authorization software *OpenAM*[10] provide ways to handle encryption issues.

2.4.2 Scalability

Scalability is a measure on how well software or hardware systems adopt on increased demands. For example, if an algorithm scales in a linear way, the calculation takes with *twice* as much computational power only *half* of the time. When talking about cloud computing people often tend to think that scalability is guaranteed and an application scales by simply moving it to the cloud. But that is clearly a myth. To achieve scalability, algorithms need to be parallelized.

Parallelization is the computational task to divide a larger problem into subproblems which

[9] Complete physical server can be provided as well.

[10] OpenAM: http://forgerock.com/openam.html.

are then solved concurrently ("in parallel") on different machines. Parallelization is the key for high performance computing and gains more importance due to the multicore paradigm of modern CPU architecture. A programming paradigm that offers parallelization is *MapReduce* (Dean and Ghemawat 2008). MapReduce is a framework invented by Google to distribute chunks of data to several machines (i.e. nodes) in a cluster. With this paradigm, every instance calculates a part of the problem and all partial results are combined at the end. This yields to a much faster execution time. The A*pache Hadoop framework*[11] provides a powerful open source implementation of MapReduce in connection with its distributed filesystem (Hadoop Distributed Filesystem, HDFS). Additionally, ambitions like the *Apache Whirr*[12] project try to simplify the set up of cluster architectures and to support the user. Finally, it is worth mentioning that it always depends on the program or algorithm itself if MapReduce suits for a given use case and yields to the desired effect. Often problems scale up to a certain point, and thereafter it does not make any sense to use more resources. And since the resources used need to be paid, it would also be a waste of money.

2.4.3 Data Transfer

IaaS is especially useful for time and data intensive calculations. One of the essential issues is how to transfer all required data into the cloud. When using a public cloud, an upload of files over the Internet is necessary. Since the upload speed of many Internet connections is often slower by a factor of 10 than the download speed, it can last a long time to upload files with gigabytes of data. For example, with an average upload speed of 70 kB/s, the transfer of 1 GB of data takes 4 h. For 1 TB (1 TB = 1,000 GB) of data the upload would already require 166 days, almost half a year. Therefore, some IaaS vendors offer the possibility to ship data on hard-drives. In order to save storage costs, only needed data should be uploaded. Compression of data is therefore one further important task which has to be considered. For example, if files mainly consist of text, they can be compressed approximately by the factor of 20 yielding to a faster upload and lower costs.

If data cannot be moved to the cloud, then one might consider of bringing the cloud to the data. Creating its own private cloud is of course more cost intensive in the beginning since an own cluster needs to be acquired and maintained. As it has been shown, it always depends on the specific case which path should be taken.

2.4.4 Usability

Usability is not only a challenge in the field of cloud computing but it is often lacking in classical software solutions. In bioinformatics one is often left alone with a command line interface and a pipeline of scripts. Additionally, setting up nodes in the cloud involves a lot of commands on the command line and some deeper understanding. Promising research approaches try to improve the usability (see Sect. 2.6.2) for end users without deeper understanding of computer science. Usability is also closely related to security, since a graphical user interface to a sensitive application is also a possible point of attack.

2.5 Cloud Computing in Action

As we learned in Sect. 2.3, cloud computing comes in different flavours. In this section the focus lies on IaaS, the possibility to access and use computer instances and thus computational power on a public cloud. This begs the question of how end users without expertise in computer science can easily access and use public clouds, and how they can use and profit from such an architecture while sitting in front of their local computer system. Using the example of Amazon EC2, we will show how end users can access IaaS easily. Table 2.1 summarises some major IaaS vendors for public and private clouds.

[11] Apache Hadoop framework: http://hadoop.apache.org.
[12] Apache Whirr project: http://whirr.apache.org/.

Table 2.1 IaaS overview (July 2011)

Type	Vendors	Comments
Public clouds	Amazon EC2, Cloud Servers, CloudSigma, Rackspace, Rightscale	Pay-per-use model for public clouds. All vendors differ in usability and business model
Private clouds	Eucalyptus	Eucalyptus is an open source platform to implement private clouds; an interface to public clouds is available ("hybrid cloud")
	vCloud Express	vCloud Express (based on VMWare) can also be used for public clouds (e.g. Terremark or Virtacore)

2.5.1 Amazon Web Services

Amazon, one of the biggest electronic commerce companies, rents its IT infrastructure in the form of Amazon Web Services (AWS) to end users. The most popular service is thereby EC2, Amazon's Elastic Compute Cloud.

This service provides the possibility to start remote computer instances in the form of virtualized machines (see Sect. 2.3.3) and access them via the Internet. To launch EC2 instances, the *AWS Management Console*[13] can be used. The AWS Management Console is a graphical web interface to set up a cluster and define hardware properties of an instance via an install wizard. Up to 20 instances can be initialized at once, whereby the number of cores, the amount of main memory, and the input/output (I/O) performance are configurable. Amazon uses a pay-per-use model, meaning that every hour a cluster instance is up and running costs a certain amount of money. The price depends on the instance type. The Management Console gives the user the possibility to gain access to remote cluster architectures or in other words to computational power in the cloud.

2.5.2 Setting Up Amazon EC2

The following steps are needed to start a cluster:

1. *Create an AWS Account*

   Similar to normal Amazon accounts personal information like name, address, email address or username are needed.

2. *Sign Up for Amazon Elastic Compute Cloud*

   Since Amazon is using a pay-per-use model, credit card information is required at this point. Each instance type provides an amount of dedicated computation capacity and is charged per instance hour consumed. The price depends on the type of instance, differentiating in the processor speed, processor cores, I/O performance, amount of main memory and storage capacity. An up-to-date pricing list is available on their website.[14]

3. *Start up a cluster using the Amazon Management Console*

   (a) A system image called Amazon Machine Image (AMI), i.e. a snapshot of a computer system containing all its data, has to be selected. This is basically the decision of using a GNU/Linux versus a Windows platform, the platform type (32 bit vs. 64 bit) and preinstalled software on the image. Different AMIs are already provided by Amazon and can be easily selected.

   (b) The amount of instances and the instance type need to be specified (see point 2). With increasing number of instances the computational power increases, and also more money has to be paid.

   (c) In a next step, a secure connection between a local computer and instances in the cloud needs to be set up. Moreover, special firewall rules can be defined and an automated process of setting up the cluster starts.

[13] Amazon AWS Management Console: https://console.aws.amazon.com.

[14] Amazon pricing list: http://aws.amazon.com/ec2/instance-types/.

2.5.3 Using Amazon EC2

When a cluster is up and running, the next logical step is to use it for processing algorithms. Two major problems arise at this point:

1. When working with cluster architectures nowadays, command line programs without graphical user interface (GUI) are often been used. Thus, installation of additional software on the instances, connecting all instances to a well-functioning distributed cluster or just the execution of a program can constitute major challenges for scientists without a computer science background. In the next section solutions are presented which try to simplify the overall setup and execution process.
2. Programs or algorithms are not automatically running faster only if they are executed in a cloud environment. The developer of a program needs to use special programming paradigms to take advantages of interconnected instances. The keyword here is parallelization and scalability (see Sect. 2.4.2).

2.6 Existing Solutions in the Field of Genetics

As mentioned in the introduction, the amount of data is growing rapidly. Thus, cloud computing constitutes an attractive alternative to deal with large datasets in adequate time and with adequate costs. In 2008 first approaches of combining bioinformatics applications and cloud computing were published and tested successfully. For these cases it was necessary to rethink underlying algorithms and to adapt them to parallel programming models like MapReduce. This turned out to be a complicated task since not every problem can be simply split into a map and reduce function.

2.6.1 Algorithmic Approaches

The pioneer of this practice was the group of Michael Schatz[15] which demonstrated with *CloudBurst* (Schatz 2009) how MapReduce can be used for the alignment of sequencing data. The program is able to align a whole genome in minutes instead of hours. Thus, small departments without own computer clusters take advantage of the possibility to perform these time-consuming computations using a public cloud. In the same year *CloudBlast* (Matsunaga et al. 2008) was developed which uses the MapReduce framework in order to parallelize the execution of NCBI BLAST2. The results showed a performance boost that can be achieved by combining technologies like cloud computing and MapReduce.

In order to fill the gap between cloud computing and usability, newer projects try to support the user by providing a simple GUI. One example therefore is *CrossBow* (Langmead et al. 2009), a scalable pipeline which can be used for whole genome resequencing. By using the MapReduce framework Hadoop, the program can be run using cluster architectures in the cloud. Another noteworthy project is *Myrna* (Langmead et al. 2010), an automatic pipeline for calculating differential gene expression in large RNA-seq datasets. Both were developed by Ben Langmead[16] and colleagues and provide a web interface that enables the execution of those algorithms in the Amazon EC2 cloud.

PeakRanger (Feng et al. 2011) is an algorithm to call peaks from ChIP-seq datasets, again using a MapReduce approach which can then be executed in a cloud environment.

2.6.2 General Approaches

Galaxy (Goecks et al. 2010) takes a completely different approach as the previously presented projects. It is not an implementation of an algorithm for a certain problem but rather a software system which facilitates the execution of existing algorithms and the creation of workflows in a fast and user-friendly way. The platform itself executes the algorithms needed for the whole analysis process step-by-step and informs the users about its progress. Galaxy's extension *CloudMan* (Afgan et al. 2010) enables installing and executing Galaxy on Amazon EC2. In a first step, the user needs to start up the master node manually by

[15] Cold Spring Harbor Laboratory in New York.

[16] Johns Hopkins Bloomberg School of Public Health.

using the AWS console. After this step the cluster can be configured via a web application where the user can dynamically add and remove worker nodes.

Elastic MapReduce (EMR) is a commercial system to execute programs graphically on Amazon's public cloud infrastructure. Since everything is located on Amazon directly, a highly optimised version of MapReduce in combination with its storage system S3 is provided and can be executed by a comprehensive user interface. Of course, Amazon Elastic MapReduce can only be used in combination with Amazon EC2, sometimes preventing research institutes from using it due to data security rules or the enormous amount of data to transfer.

A further approach is the free software system *Cloudgene*,[17] which simplifies the access to computational resources and associated computational models of cluster architectures, assists end users in executing and monitoring developed algorithms via a web interface and provides an interface to add future developments or any kind of programs.

Another popular possibility is producing special system images with preinstalled software and metadata on it. A mentionable project is *CloudBioLinux*,[18] a system image for Amazon EC2 or Eucalyptus with preinstalled biological software, programming libraries and data sets. The free available image can be started on an Amazon EC2 instance and provides the possibility to work on it via a graphical remote desktop. The installed software can be used for several tasks like aligning and statistical analysis.

2.6.3 Data Management

Beside software tools that benefit from the computational power and the scalability of a cloud, systems for data management were developed. On top of the cloud infrastructure several laboratory information management systems (LIMS) were designed. One example therefore is the *SeqWare Query Engine* (O'Connor et al. 2010), a system with a web-based frontend for storing and searching thousands of next-generation sequencing data.

Uploading huge data sets (e.g. the mapping of the human genome) requires hours or days until they are copied to the cloud (see Sect. 2.4.4). To eliminate this bottleneck, Amazon hosts a lot of *public datasets*[19] on their S3 storage system which are accessible for every AWS customer. This service removes time intensive uploading processes and the same datasets can be reused by other researchers as well. A lot of research institutes take up this idea and share also their data and results through this service. Examples for popular datasets are the annotated human genome, *GenBank*,[20] *HapMap*[21] or *UniGene*.[22]

2.7 Summary

This chapter gave an overview of cloud computing in general to pass a basic understanding of technologies, challenges and opportunities of this hot topic in bioinformatics. Special focus was on a practical understanding of IaaS to communicate the reader the idea on how to use cloud computing approaches for own research projects efficiently.

References

Afgan E, Baker D, Coraor N, Chapman B, Nekrutenko A, Taylor J (2010) Galaxy CloudMan: delivering cloud compute clusters. BMC Bioinformatics 11(Suppl 12): S4. doi:10.1186/1471-2105-11-S12-S4

Clark J (2011) NHS laptop loss could put millions of records at risk. http://www.zdnet.co.uk/news/security-management/2011/06/15/nhs-laptop-loss-could-put-millions-of-records-at-risk-40093112/. Accessed 20 Jun 2011

Dean J, Ghemawat S (2008) Mapreduce: simplified data processing on large clusters. Commun ACM 51(1):107–113

[17] Cloudgene: http://cloudgene.uibk.ac.at.

[18] CloudBioLinux: http://cloudbiolinux.org.

[19] Public Datasets on Amazon: http://aws.amazon.com/publicdatasets.

[20] GenBank: http://www.ncbi.nlm.nih.gov/genbank/.

[21] HapMap: www.hapmap.org.

[22] UniGene: http://www.ncbi.nlm.nih.gov/unigene.

Feng X, Grossman R, Stein L (2011) PeakRanger: a cloud-enabled peak caller for ChIP-seq data. BMC Bioinformatics 12:139. doi:10.1186/1471-2105-12-139

Goecks J, Nekrutenko A, Taylor J (2010) Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome Biol 11(8):R86. doi:10.1186/gb-2010-11-8-r86

Holland R (2011) Ten steps to successful cloud migration. http://www.eaglegenomics.com/download-files/whitepaper/CloudWhitePaper.pdf. Accessed 20 Jun 2011

Langmead B, Schatz MC, Lin J, Pop M, Salzberg SL (2009) Searching for SNPs with cloud computing. Genome Biol 10(11):R134. doi:10.1186/gb-2009-10-11-r134

Langmead B, Hansen KD, Leek JT (2010) Cloud-scale RNA-sequencing differential expression analysis with Myrna. Genome Biol 11(8):R83. doi:10.1186/gb-2010-11-8-r83

Markovich S (2010) How to secure sensitive data in cloud environments. http://www.eweek.com/c/a/Cloud-Computing/How-to-Secure-Sensitive-Data-in-Cloud-Environments/. Accessed 20 Jun 2011

Matsunaga A, Tsugawa M, Fortes J (2008) Cloudblast: combining mapreduce and virtualization on distributed resources for bioinformatics applications. In: Proceedings of the 2008 fourth IEEE international conference on eScience, IEEE, pp 222–229. doi:10.1109/eScience.2008.62

O'Connor BD, Merriman B, Nelson SF (2010) SeqWare query engine: storing and searching sequence data in the cloud. BMC Bioinformatics 11(Suppl 12):S2. doi:10.1186/1471-2105-11-S12-S2

Rittinghouse J, Ransome J (2009) Cloud computing: implementation, management, and security, 1st edn. CRC, Boca Raton

Schatz MC (2009) CloudBurst: highly sensitive read mapping with MapReduce. Bioinformatics 25(11):1363–1369. doi:10.1093/bioinformatics/btp236

Wetterstrand KA (2011) DNA sequencing costs: data from the NHGRI large-scale genome sequencing program. http://www.genome.gov/sequencingcosts. Accessed 11 Apr 2011

http://www.springer.com/978-3-7091-0946-5

Computational Medicine
Tools and Challenges
Trajanoski, Z. (Ed.)
2012, IX, 203 p., Hardcover
ISBN: 978-3-7091-0946-5