NIBIO
NORWEGIAN INSTITUTE OF BIOECONOMY RESEARCH

Spatial Big Data tools and methods

A guide to possible ideas, new opportunities and challenges for NIBIO

NIBIO REPORT | VOL. 7 | NO. 157 | 2021

Jonathan Rizzi, Misganu Debella-Gilo
Division of Survey and Statistics, Department of Geomatics

TITTEL/TITLE

Spatial Big Data tools and methods within NIBIO

FORFATTER(E)/AUTHOR(S)

Jonathan Rizzi, Misganu Debella-Gilo

DATO/DATE:	RAPPORT NR./ REPORT NO.:	TILGJENGELIGHET/AVAILABILITY:	PROSJEKTNR./PROJECT NO.:	SAKSNR./ARCHIVE NO.:
21.10.2021	7/157/2021	Open	11028	19/00827

ISBN:	ISSN:	ANTALL SIDER/ NO. OF PAGES:	ANTALL VEDLEGG/ NO. OF APPENDICES:
978-82-17-02918-2	2464-1162	41	

OPPDRAGSGIVER/EMPLOYER:	KONTAKTPERSON/CONTACT PERSON:
NIBIO	Jonathan Rizzi

STIKKORD/KEYWORDS:	FAGOMRÅDE/FIELD OF WORK:
Stordata	Geomatikk, IKT
Big Data	Geomatics, ICT

SAMMENDRAG/SUMMARY:

Rapporten utforsker og diskuterer potensialet for økt bruk av *Stordata* (engelsk: *big data*) teknologi og metode innenfor instituttets arbeidsområder. I dag benyttes *Stordata*-tilnærminger til å løse forvaltningsstøtteoppgaver, samt til forskningsformål, særlig i sentrene for presisjonslandbruk og presisjonsjordbruk. Potensialet for økt bruk av *Stordata* innenfor instituttet er stort. For å realisere potensialet er det behov for god samordning mellom organisasjonsenhetene og utvikling av strategisk kompetanse på fagområdet.

LAND/COUNTRY:	Norway
FYLKE/COUNTY:	Viken
KOMMUNE/MUNICIPALITY:	Ås
STED/LOKALITET:	Ås

GODKJENT /APPROVED	PROSJEKTLEDER /PROJECT LEADER
Hildegunn Norheim	Jonathan Rizzi
NAVN/NAME	NAVN/NAME

NIBIO
NORWEGIAN INSTITUTE OF BIOECONOMY RESEARCH

Preface

Most activities of NIBIO require that geospatial information of various type is available or generate new spatial information as output. The organization handles large amounts of data, and new information is generated as output leading to a rapid increase of the volume of data. It is therefore vital for NIBIO to have the most advanced competences to deal with geospatial big data. In 2018 NIBIO started an internal competence project to build knowledge in this sector across all the divisions of the institute.

NIBIO is already using big data technology and methodology in several departments, and particular in the centres for precision agriculture and the centre for precision forestry. These centres have researches with competences on the subject. However, until 2018 there was no coordination and little communication across divisions, which is leading to limited exploitation of the huge potential already available in NIBIO as well as a loss of efficiency due to replication of activities, data, or processes. The *Stordata* project, hosted by the Geomatics department under the Survey and Statistics Division is aiming to better link people and activities for the benefit of the entire NIBIO organization.

The aim of this document is to present the first outcomes of the project and provide suggestions and possible pathways to guide NIBIO further into the world of geospatial big data.

Ås, 21.10.21

Hildegunn Norheim

Content

1 Introduction

The importance of a well-functioning bioeconomy is increasingly recognised in addressing challenges such as food safety, natural resource scarcity, climate change, unsustainable development, and consumption patterns. Defined as an economy in which food, materials and energy are derived from renewable biological resources involving the land and the sea (Commission, 2012), bioeconomy is seen as a central component of sustainable development. Availability of data and information is crucial to be able to take correct decisions at all levels and in many other sectors, from healthcare to transport and energy, including bioeconomy, big data have become fundamental.

There are multiple definitions of big data depending on the perspective on which it is defined. According to the European Union[1], the term big data refer to "large amounts of data produced very quickly by a high number of diverse sources. Data can either be created by people or generated by machines, such as sensors gathering climate information, satellite imagery, digital pictures and videos, purchase transaction records, GPS signals and more". S. Li et al. (2016) defined big data concisely as "structured and unstructured datasets with massive data volumes that cannot be easily captured, stored, manipulated, analysed, managed and presented by traditional hardware, software and database technologies." Three fundamental characteristics are consistently mentioned in the various definitions of big data. These are Volume, Velocity and Variety, often referred to as the three V's of big data. Additionally, in numerous cases of big data definition Veracity and Value are also included as big data characteristics, extending the three V's to five V's.

- **Volume:** the quantity of generated and stored data. The size of the data determines the value and potential insight, and whether it can be considered big data or not. It is widely acknowledged that the world is being overwhelmed by the sheer volume of data which is increasing exponentially with time in every sector (Hilbert & Lopez, 2011).
- **Velocity:** the speed at which the data is generated, ranging from periodic and batch acquisition to real time data streaming, and processed to meet the demands and challenges that lie in the path of growth and development. Big data is often produced continuosly and available in real-time or near-real time. Two kinds of velocity related to big data are the frequency of generation and the frequency of handling, recording, and publishing.
- **Variety:** the type and nature of the data. This helps people who analyse the data to effectively use the resulting insight. Big data draws from text, images, audio, video; plus it completes missing pieces through data fusion.
- **Veracity:** the data quality and the data value. The data quality of captured data can vary greatly, affecting the accurate analysis.
- **Value:** data itself and processed data has a value that can be quantified directly or indirectly in monetary terms.

The new challenges coming from data complexity and the demands for faster information management, elaboration and extraction call for a paradigm shift in the way data is acquired, stored, processed, and analysed at NIBIO. The solution for the challenges lies in the methods and in the technologies, platforms, and software solutions of big data, all of them advancing rapidly. Specifically, the geospatial part of big data solution is a technology in rapid progress. For a comprehensive solution, the whole information value chain must be traversed in relation to the big data technologies, platforms, and software and in relation to the local and national capacity. Capacity building in geospatial big data,

[1] https://ec.europa.eu/digital-single-market/en/big-data

specifically in terms of competence and infrastructure, is a key to efficient data driven information extraction and decision support.

1.1 The *Stordata* project

The *Stordata* (*Stordata* is the Norwegian term for big data) project was initiated to increase the competence within NIBIO and take the benefits of the application of new methods, such as machine learning, in the daily activity of the institute. The project started in 2018 and was initially organised in three main Work Packages (WPs):

- **WP1: Alliance and networking.** This work package is mainly related to the establishment of connection with relevant actors both in Norway and abroad.
- **WP2: Machine learning and remote sensing for automatic map changes.** This work package is mainly related to the identification and testing of methods that can be relevant for NIBIO and will focus in particular on methods that can be useful for the updated of AR5 dataset2.
- **WP3: Technology and platforms for convenient big data with geographic data.** This work package is mainly related to the analysis and testing of different solutions for storing and processing data (including, e.g, use of distributed computing, use of HPC or other infrstrastrucutre for data storage/processing such as Uninett Sigma2).

In 2019 the Division of Biotechnology and Plant Health participated in the project with an additional work package. The work has not been focusing on geospatial big data but on use of big data in image analysis. The work included analysis related to the following three cases.

- **Case 1:** use of artificial intelligence to improve existing warning models in VIPS (Varsling Innen PlanteSkadegjørere).
- **Case 2:** use of machine learning, specifically non-guided learning, to identify any relationships between environmental factors and outbreaks of a pest.
- **Case 3:** use of artificial intelligence for automatic identification and quantification of plant pests from images, with two areas: i) automatic identification of plant pests (fungal spores / potato wire) from images taken in the microscope; ii) automatically find the position and distribution of weeds that occur simultaneously in the same culture.

In the same year, NIBIO became partner of the project NFR 295836 - *Maskinlæring for automatisk kartlegging av kommunale FKB- og temadata basert på laser og hyperspektrale data* (Machine learning for automatic mapping of municipal FKB and topic data based on laser and hyperspectral data). The participation of NIBIO is mainly as observer and is included within the activities of WP2.

1.2 The *Stordata* project team

The *Stordata* project involved several people from different divisions/departments, as listed hereafter (following alphabetical order of family name).

- **Therese With Berge** – Division of Biotechnology and Plant Health – Invertebrate Pests and Weeds in Forestry, Agriculture and Horticulture
- **Misganu Debella-Gilo** – Division of Survey and Statistics – Department of Geomatics

[2] AR5 is a land resource map at a scale of 1: 5000. AR5 is a detailed, nationally comprehensive data set and the best source of information on the country's land resources. The data set divides the land by area type, forest quality, tree species and soil conditions.

- **Håvard Eikemo** – Division of Biotechnology and Plant Health – Department of Fungal Plant Pathology in Forestry, Agriculture and Horticulture
- **Brita Linnestad** – Division of Biotechnology and Plant Health – Department of Fungal Plant Pathology in Forestry, Agriculture and Horticulture
- **Åsmund Ertshus Mathisen** – Division of Survey and Statistics – Department of Land Inventory
- **Ingvild Nystuen** – Division of Survey and Statistics – Department of Geomatics
- **Jonathan Rizzi** – Division of Survey and Statistics – Department of Geomatics
- **Anne-Grete Roer Hjelkrem** – Division of Food Production and Society – Department of Agricultural Technology and System Analysis
- **Tor-Einar Skog** – Division of Biotechnology and Plant Health – Department of Fungal Plant Pathology in Forestry, Agriculture and Horticulture
- **Ingerd Skow Hofgaard** – Division of Biotechnology and Plant Health – Department of Fungal Plant Pathology in Forestry, Agriculture and Horticulture
- **Nils Egil Søvde** – Division of Survey and Statistics – Department of Geomatics
- **Jiangsan Zhao** – Division of Biotechnology and Plant Health – Viruses, Bacteria and Nematodes in Forestry, Agriculture and Horticulture

2 Big data in NIBIO

In NIBIO, there is a fast increase of the total volume and rate of inflow of data from various sources. Some of the data are generated by the institute itself while others are acquired from other sources including national and international organizations. The type of data is heterogenous in term of format, file dimension, quality, and accuracy. Further, there is often redundancy in the data that is stored and processed at the institute. Storing and analysing these datasets require both high computational and storage capacity.

A large part of the geospatial data used at NIBIO is coming from remote sensing and sensor technologies. An example of large archive is represented by the microwave (Sentinel-1) and optical (Sentinel-2) images from the ESA's Copernicus satellite series. Images are downloaded and analysed at NIBIO for different purposes. Sentinel-2 images covering entire Norway with only one acquisition have a disk dimension of around 90 GB. A set of images covering the whole country is generally acquired over a period of ~5 days, and continues to grow with the same rate, i.e. 90 GB every 5 days. Consequently, optical images over all Norway acquired between 2016 and 2018 by the ESA's Setinel-2 satellite amounts to over 15 TB. There are several satellite datasets currently in use at NIBIO for different purposes to preform analysis from the local to the global scale, such as Landsat or MODIS optical images of USGS/NASA or radar images such as Tandem-X of the DLR. As an example, the global coverage of TandDem-X data amounted to over 2 TB.

Another large part of the geospatial data are aerial images taken from airborne optical or radar sensors. Norway undertakes aerial photogrammetric campaigns over large areas every year with a repeating period of five to seven years, generating large volumes of data. In addition, a national aerial laser scanning campaign has recently been completed covering the entire land area of the country, generating big and complex point cloud data. Based on these data, other datasets are created, such as digital elevation models, generating even more data. NIBIO is one of the major Norwegian institutes that use both the raw point clouds and the derived attributes on a national scale.

Other large geospatial datasets in use at NIBIO, and rapidly increasing in terms of volume, are weather/climate models' outputs. More and more researches are nowadays related to weather and/or climate requiring data related to multiple variables and covering large areas at high resolution and over long time series. It is not unusual that those data are replicated within NIBIO.

The increase of data volume is due also to recent technological developments such as the diffusion of sensor networks for the real time monitoring of different environmental parameters. In NIBIO networks of sensors are mainly operated, for e.g., by researchers interested in water resources monitoring or in the weather monitoring in relation to pests monitoring and risk assessment.

Technological improvements and lower costs of UAV (Uncrewed Aerial Vehicles) in another example of development leading to the generation of very relevant data that need to be stored and processed. UAV data is particularly relevant in sectors such as forestry and agriculture. These sectors make use of different type of machines (e.g., harvesters, tractors, etc.): sensors and logging machines mounted on those machines can provide quite useful geospatial data.

A last example of geospatial datasets is represented by socio-economic data that are collected by individual farmers, municipalities, and the state for various purposes. These data that once were mainly only alphanumerical, are becoming more and more georeferenced and spatially explicit, opening for more advanced analysis and research activities.

Beyond these examples, it is also important to remember that any dataset that is analysed, generates large volumes of intermediate data. For example, images generated by drones are processed to produce digital surface models (DSM), digital terrain models (DTM) and orthographic images (orthophoto). The volume of these derived data is at least as large as the original image size itself. Derived products need

to be stored and processed further to extract other useful information. The whole information value chain is challenged by this ever-increasing complexity of data. As one of the outputs of these activities is the publication of scientific paper, it is important to remind that data related to scientific publications must be kept and be available for at least ten years from the date of publications of the paper. It is not therefore possible to delete data that sometimes seems not useful anymore and is important to have solution also to store these datasets. A solution for publishing final datasets also allowing the attribution of DOI, called BIRD[3], has been implemented by a consortium as several institutions/universities and is already in use at NIBIO.

A more comprehensive list of the geospatial big data related to different NIBIO divisions is given below. The list is dynamic and, therefore, cannot be considered as final.

- Division of Food Production and Society
    - Optical remote sensing data (from satellites to UAV)
    - Laser remote sensing data (from satellites to UAV)
    - Sensor data from machines used in the agricultural sector
- Division of Environment and Natural Resources
    - Sensor data related to water resources (from sampling campaigns or from fixed instruments)
    - Sensor data related to soil monitoring
    - Weather/climate data
- Division of Forest and Forest Resources
    - Optical remote sensing data (from satellites to UAV)
    - Laser remote sensing data (from satellites to UAV)
    - Sensor data from machines used in the forestry sectors
    - Climate/weather data
    - National Forest Inventory
- Division of Biotechnology and Plant Health
    - Weather/climate data
    - Images from microscope (not geospatial but still images)
- Division of Survey and Statistics
    - Socio-economic data from farms and national registers
    - Optical remote sensing data (from satellites to UAV)
    - Laser remote sensing data (from satellites to UAV)
    - Weather/climate data
    - Data related to monitored species, e.g., of plants and birds.
    - Pictures from standard cameras

It is evident that some data sources can be relevant for several divisions, it is therefore important to have a system to share information and optimize data management and storage, in order to avoid replication of data and its analysis, and enhance data sharing among the users .

[3] https://bird.unit.no/resources/search?institution=nibio

3 The big data value chain

A value chain is a model of the chain of activities that an organisation performs to deliver a product or a service to the market. The value chain identifies the main activities, allowing them to be understood and optimised. The way data (and therefore also geospatial big data) is managed within an organization from its acquisition to its final use can be modelled using a value chain to understand the value-creation of data technologies.

Figure 1. The data value chain.

The data value chain, as illustrated in Figure 1Figure 1. The data value chain., models the high-level activities that comprise an information system. The data value chain identifies the following activities:

- **Data Acquisition** is the process of gathering, filtering, and cleaning data before it is put in a data warehouse or any other storage solution on which data analysis can be carried out.
- **Data Curation** is the active management of data over its life-cycle to ensure it meets the necessary data quality requirements for its effective usage.
- **Data Storage** is concerned about storing and managing data in a callable way satisfying the needs of applications that require access to the data.
- **Data Retrieval** is concerned with efficient detection and retrieval of data from the storage systems.
- **Data Analysis** is concerned with making raw data, which has been acquired, amenable to use in decision-making as well as domain specific usage.
- **Data Usage** covers the business goals that need access to data and its analysis, and the tools needed to integrate analysis in business decision-making.

The elements of the data value chain have a common element: they require access to (or transmission of) data. Data access is realized through connection and transfer of data between a computer storage medium and its RAM, between nodes of cluster racks, between computers connected over a network, etc. Data access costs time, the amount depending on the connection characteristics. Data transmission can consequently create bottlenecks in big data solutions (Chen et al., 2014).

When data is stored on local storage media with large capacity such as the DAS (Direct-Attached Storage) system, data is accessed by the disk I/O. Although this removes the need for network traffic, it lacks the flexibility and scalability of data access. On the other hand, when data is stored in networked media such as the NAS (Network-Attached Storage) and SAN (Storage Area Network), bandwidth is required for data transmission. High speed network systems such as fiber optics are crucial in avoiding bottlenecks in big data solutions. Yet, the transfer of data to HPC clusters and cloud computing centres is the major bottleneck in remote computing.

The big data paradigm entails that data transmission is taken over by script transmission. However, there is still a need to store the data in the first place and retrieve the processing results. Many technical and infrastructure solutions are proposed to circumvent the problem of latency in data transmission. Efficient algorithm to filter and compress data, near-source pre-processing with the use of mist or fog computing architecture, etc.

The different elements of the data value chain will be discussed in the following sections focusing on geospatial big data as type of information processed by the chain.

Data Acquisition

Geospatial big data are nowadays generated by numerous technologies. The major sources of traditional geospatial big data are field surveys and remote sensing technologies such as satellites and airborne photogrammetry, radar and lidar technologies. There are also ground-based remote sensing technologies. Relatively recent technological developments such as GPS, mobile phones, UAVs, Internet of Things (IoT) through sensor networks, etc. are generating enormous amount of geospatial data. The raw data are generated either in batch forms or near-real time or real time data streams.

Data acquisition is the ensemble of the pre-processing operations such as gathering, filtering, and cleaning data before it is stored. Data acquisition is one of the major big data challenges in terms of infrastructure requirements. The infrastructure required to support the acquisition of big geospatial data must deliver low and predictable latency in both capturing data and in executing queries, be able to handle very high transaction volumes, often in a distributed environment, and support flexible and dynamic data structures.

An important aspect of data acquisition, data integration, is dealing with ingesting generated data into the data storage system in the form of batch processes and data streaming. A data ingestion procedure includes an input unit represented by the data source, a process unit where data are checked and filtered, and an output unit where the data are stored. This task becomes challenging in the case of streaming data where the whole pace of data acquisition (ingestion) should keep up with the pace of data generation. Latency is therefore a crucial factor. A number of ready-made programs are available depending on the data type one is dealing with (Chen, Mao, & Liu, 2014; Lyko, Nitzschke, & Ngonga Ngomo, 2016a) and tools used for ingestion of big data into data storage system such as Sqoop or Flume (Lee & Kang, 2015; Lyko, Nitzschke, & Ngonga Ngomo, 2016b) are among the most popular ones.

Data Curation

The active management of data over its life cycle to ensure it meets the necessary data quality requirements for its effective usage is the focus of data curation (Pennock, 2007). Data curation processes can be categorised into different activities such as content creation, selection, classification, transformation, validation, and preservation.

Data curation is performed by expert curators that are responsible for improving the accessibility and quality of data. Data curators (also known as scientific curators, or data annotators) hold the responsibility of ensuring that data are trustworthy, discoverable, accessible, reusable, and fit their purpose. A key trend for the curation of big data utilises community and crowd sourcing approaches (Curry et al. 2010).

Data Storage

Data storage includes the persistence and management of data in a scalable way that satisfies the needs of applications that require fast access to the data. Relational Database Management Systems (RDBMS) have been the main, and almost unique, solution to the storage paradigm for nearly 40 years. However, the ACID (Atomicity, Consistency, Isolation, and Durability) properties that guarantee database transactions lack flexibility regarding schema changes and the performance and fault tolerance when data volumes and complexity grow, making them unsuitable for big data scenarios. NoSQL (Not Only SQL) technologies have been designed with the scalability goal in mind and present a wide range of solutions based on alternative data models.

From a physical point of view, there is a wide range of storage media with varying capacity and efficiency. They can be expressed in terms of storage technology or network architecture (Hu, Wen, Chua, & Li, 2014). Data can be stored in Random Access Memory (RAM) which is lost after the system is powered

off. This is only used as a data buffering mechanism during data processing and analysis. Magnetic disks and disk arrays are common forms of data storage in the form of Hard Disk Drives (HDDs). Traditional HDDs have fast moving parts which are prone to failure requiring robust protection. The RAID (Redundant Array of Independent Disks) that accompanies traditional HDDs is not flexible enough to scale up to the volume and velocity of big data (Yang, Huang, Li, Liu, & Hu, 2017). Non-mechanical storage media such as Flush memory that are used to construct Solid State Disks (SSDs) have no moving parts and are faster in retrieving data, but they are costly. A compromising combination of the different storage media could be set up for optimization of cost and reliability.

Data Retrieval

Metadata, documentation, and intelligent organization is necessary to facilitate and ensure that data can be retrieved from the storage media in a timely and efficient way. Large data volumes are more demanding in this respect. Additionally, the retrieval of geospatial data often involves geometrical or topological operations where the organization of data on the storage media has considerable impact on the performance and efficiency on the operations.

Data Analysis

The next step in the data value chain is the data analysis, which is related to making the raw data acquired amenable to use in decision-making as well as domain-specific usage. Data analysis involves exploring, transforming, and modelling data with the goal of highlighting relevant data, synthesising, and extracting useful hidden information with high potential from a business point of view. This relates to Artificial Intelligence in a general way and with machine learning and deep learning. It is important to clarify these terms since often there is some misuse of them. Their interrelations and hierarchy are also often not clear. Machine learning is a set of methods that are within the field of artificial intelligence and deep learning include several methods as a subset of machine learning.

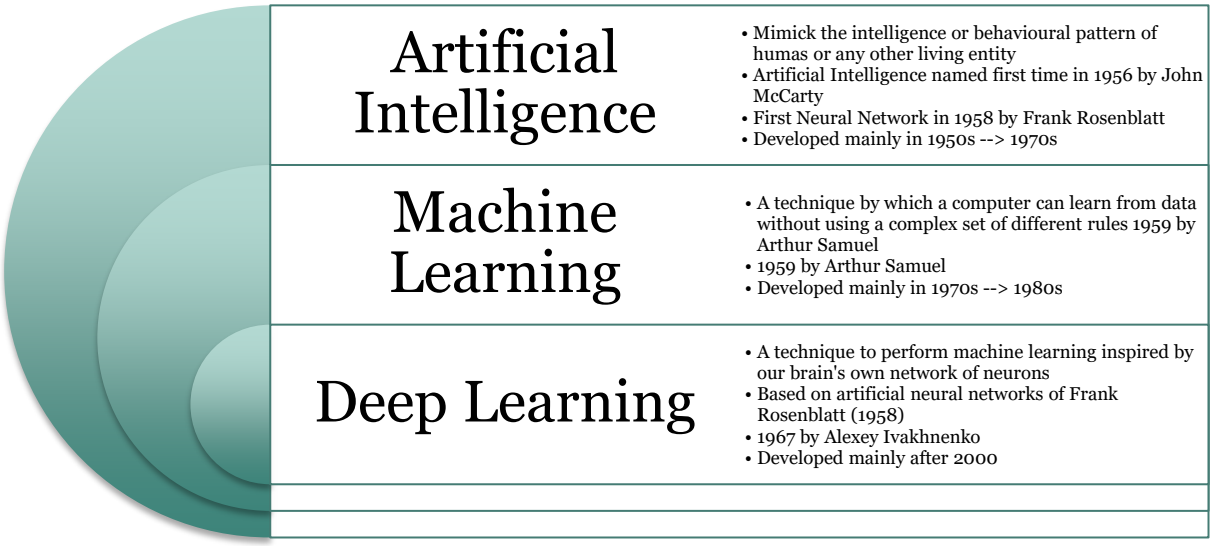

Figure 2. Hierarchy and short definition of artificial intelligence, machine learning and deep learning.

- **Artificial intelligence (AI):** is the intelligence demonstrated by machines, unlike the natural intelligence displayed by humans and animals. AI is the study of "intelligent agents": any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals. The term is often used to describe machines (or computers) that mimic cognitive functions that humans associate with the human mind, such as learning and problem solving.
- **Machine Learning (ML):** is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence. Machine learning algorithms build a

mathematical model based on sample data, known as training data, or patterns within the data to make predictions or decisions without being explicitly programmed to do so.

- **Deep [structured] Learning (DL):** is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised. Deep learning uses multiple layers to progressively extract higher level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces.

Data Usage

The last step of the big data value chain is represented by data usage, covering the data-driven business activities that need access to data, its analysis, and the tools needed to integrate the data analysis within the business activity. Data usage in business decision-making can enhance competitiveness through reduction of costs, increased added value, or any other parameter that can be measured against existing performance criteria.

The Big Data value chain in NIBIO

To improve the efficiency, it is worth having an overview of how the big data value chain can be implemented within NIBIO, referring as always to spatial big data. All divisions dealing with such data have researchers with competences on big data, and it is not a good solution to centralize everything related to big data in a single unit. However, it is important to have some coordination, especially on some elements. A special role can be played by the Geomatics department, which can act as central point to coordinate initiatives and give technical support to other divisions.

Table 1. Activities related to the steps of the big data value chain implemented in NIBIO.

Step of the value chain	Activity at the Department of Geomatics/ICT	Activities at all other divisions
Data Acquisition	• Provide support for the infrastructure and help finding the optimal solution • Keep an overview of external datasets of common interest (e.g., satellite data, climate data) and share information to avoid data replication	• Collect data • Download datasets of interests following standard guidelines • Share metadata
Data Curation	• Provide technical support/assistance	• Perform necessary activities
Data Storage	• Provide the storage infrastructure based on the needs of the different divisions • Provide training on the storage procedures in use at NIBIO and external facilities (such as HPC and cloud)	• Use the storage infrastructure following standard guidelines • Build capacity to be able to use internal and external data storage facilities
Data Retrieval	• Provide and support for improved data retrieval • Information about metadata standards • Provide expertise on topology	• Documentation • Metadata creation • Intelligent organization of data
Data Analysis	• Provide training on topics of common interest across divisions • Organize a network of people working with big data • Perform big data analysis	• Perform big data analysis • Build capacity to be able to use internal and external data analysis and computational facilities
Data Usage	• Provide support on demand	• Perform necessary activities

4 Tools for Big Data Storage and Processing

The big data paradigm entails that a single personal computer is not capable of retrieving and processing the data. Traditional serial computation where the actions taken on a given data is performed one after the other or an action is taken on numerous data one after the other is inefficient in this era of big data. Even traditional parallel processing schemes may not be capable of handling big data, particularly distributed data such as the HDFS (Hadoop Distributed File System, see below for more details). The actions to be taken should have the possibility of performing many tasks at the same time, besides there should be a possibility of performing a task on multiple data at the same time. The computational architecture therefore must reflect the data storage and management system.

The major paradigm shift is that instead of taking data to the software or the script, the script is sent to the data, reducing the need for data transmission and redundant storage. Both distributed data storage and distributed computation is central to big data. The computational architecture can be divided into the hardware architecture and the software framework.

Hardware architecture

As the challenges of big data are specific to each enterprise or institute, no single architecture suits all challenges. Different hardware configurations can be designed depending on the challenges and requirements. Smaller to medium data problems can, for example, be solved by using super computers with powerful CPUs (and GPUs) in addition to a larger storage media. However, as the data volume and the requirements for acquisition or processing velocity increases, supercomputers may not be sufficient. Networks of such computers in a distributed computing framework can be used in such cases. In an even larger challenge, the solution may require establishment of a computing centre with High Performance Computing (HPC) infrastructures which can be used as a storage and computing host. This can be organized at institute or national level, for example Sigma2 of UniNett. Nowadays there are numerous commercial Cloud computing services that offer complete infrastructure from data storage to computing and analysis, together with scripting capabilities. Most of these solutions rely on data transmission broadband which may create a bottleneck especially in a high velocity data acquisition and data processing scheme. To circumvent such problems in high velocity data acquisition such as sensor-based data streaming, intermediate solutions between the data source and HPC or cloud services can be designed. These can be smaller, but numerous, computing devises that pre-process the collected data and relay it to the storage centre. Such architecture is sometimes referred to as mist computing or fog computing (Iorga et al., 2018; Uehara, 2017).

Software framework

As stated above, big data requires a computational framework that fits into the data storage model and the characteristics of the data. There is a long list of such solutions out there and many more might also be in the process of development. Here, we discuss a few of those we thought are widely used in the areas of geospatial big data.

In this section there will be an introduction to some tools for data management and data storage. While many tools started with specific purposes, i.e., either data management or data storage, nowadays it is more and more common to have solutions that support the implementation of several parts of the big data value chain.

4.1 Solution and format for big data storage

For efficient processing and analysis, data need to be stored, managed, and accessed with high reliability and efficiency. Data storage has three components: i) disk arrays for storage; ii) connection and network for data transmission; and iii) storage management software (Chen et al., 2014). For efficient data storage (and consequently also for an efficient data sharing), there should be no bottlenecks in these

three components. In the following subsections, some relevant solutions for big (spatial) data will be presented.

Data management software

The capability of storing the data alone is not sufficient for fully functional big data solution. The data must be accessible (preferably concurrent access), ready for parallel processing, resilient against failure, etc. Therefore, in addition to the hardware storage capacity, data management software is crucial in big data storage. Traditional file-based systems that store and manage data with no awareness of the contents and relationship between the files is not optimal for big data. Database Management Systems (DBMS) are well suited for structured data and deal with much of the weaknesses of the file-based data management systems. However, as data become bigger and more complex, the need for a data management system capable of handling structured as well as unstructured data with possibilities of distributed processing and resilience against failure is needed. Both file-based and RDBMS lack this flexibility and capability to variable extent. Additionally, data management systems based on programming models are also available.

From Google File System to Hadoop Distributed File System

Google published a paper on Google File System (GFS) in 2003 (Ghemawat et al., 2003) which extends the file-based data management to distributed file system. The technique gives possibilities to break down big files into smaller pieces so that it can be handled by smaller memories and duplicates the storage so that it is resilient against failure. Additionally, the method provides possibility for concurrent access. Big companies such as Apache have developed it further and produced a data storage platform called the HDFS, Hadoop Distributed File System (Shvachko et al., 2010). Hadoop is a fully developed ecosystem of data acquisition, storage, management, processing, and analysis. The HDFS component of Hadoop is widely used by enterprises for storage of structured and unstructured big data. HDFS is highly scalable, fault tolerant and suitable for distributed computing. Although it is widely implemented by large enterprises, it can easily be implemented also by smaller ones as it is readily scalable to any size.

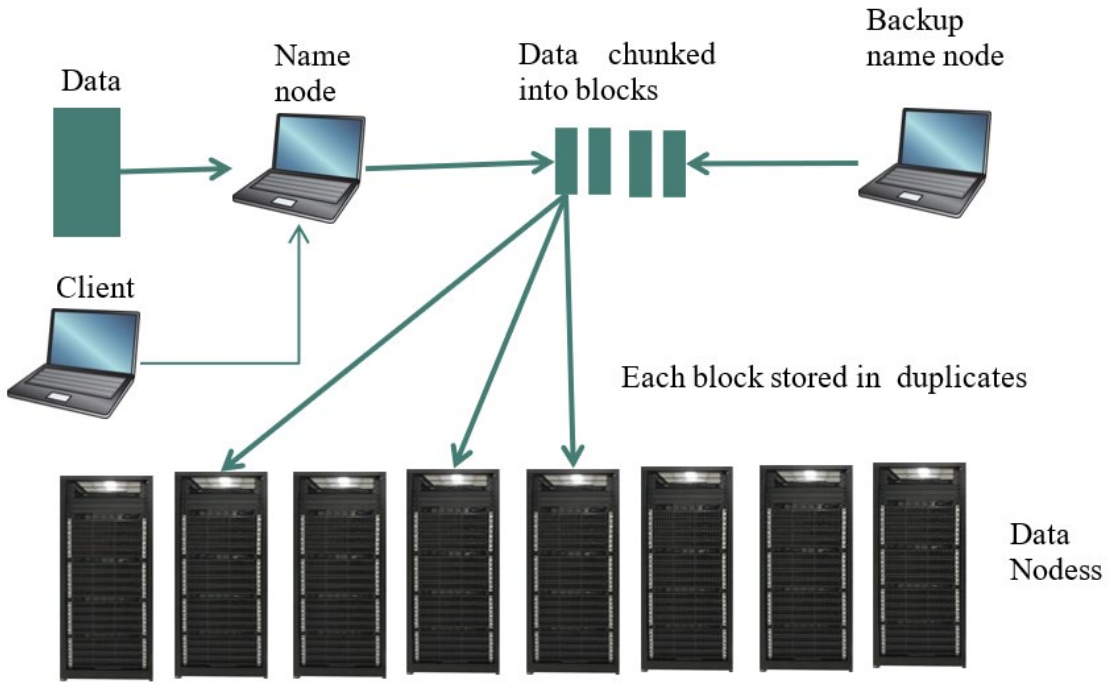

Figure 3. The Hadoop HDFS architecture.

Figure 3 depicts the architecture of the HDFS. In the system, one machine, called Name node, and preferably an additional one for backup called Backup Name node, is used as a bookkeeper. Its tasks are dividing big files into blocks, sending each block in specified number of duplicates to the storage media, keeping records of files, blocks, and their storage location, following the heartbeat of the storage media. Multiple data nodes, preferably thousands, store the blocks and process them whenever a processing action is requested and sends heartbeat, storage status, processing result, etc. back to the name node. Any client machine intending to work on the HDFS connects to the name node and sends tasks to the name node which will then distribute and follow up the tasks to eventually provide feedback to the client.

The HDFS has some limitations: i) the redundancy mode increases cumulative data size; ii) there is a weak possibility of querying; iii) spatial aspect is a specialized add-on and it is still not fully developed; and iv) it requires special programming scheme to process data on the HDFS although other programming schemes such as the Apache Spark are later developed.

Testing of HDFS in NIBIO

A test run using connected personal computers showed that HDFS can be implemented at the local level. Three personal computers with total RAM capacity of 48 GB were linked. All three computers run on Windows 7 operative system. They are connected through port 22 as recommended and defaulted in the Hadoop manual.

Hadoop is originally developed for the Linux operative system. Installing and running Hadoop in a distributed system on windows is not straight forward. There was no problem to install and run single-node Hadoop on a single machine. However, the cluster version requires work around to run it on Windows. The solution is the use of CYGWIN that mimics the Linux operative system. CYGWIN is installed and configured in a similar fashion on all the three computers. The Windows version of Hadoop is then compiled from source following the guidelines given at the apache website (https://cwiki.apache.org/confluence/display/HADOOP2/Hadoop2OnWindows). The installed windows version is then copied to the two other computers.

The installed Hadoop ecosystem has the components called HDFS for distributed file storage, MapReduce for distributed processing and Yarn for resource management. The HDFS is tested to store data in a distributed format on the three nodes. The HDFS system requires that at least one node is used as the name node keeping track of the files, their distributed blocks and where they are stored, i.e., bookkeeping. Then the other two or all the three are used as data nodes for storing the data in a distributed fashion.

A file with a size of 7.5 GB was then loaded on to the HDFS by the name node. The file is broken down into pieces and stored in two replicas distributed over the three computers, as is possible to see by looking at the book-keeping by each name node. This is just a simple demonstrative test of storing a file in the HDFS system.

NoSQL (Not only Structured Query Language)

Where RDBMS is limited to structured data and lack the horizontal scalability and distributed storage that is needed for distributed computing (discussed below), the HDFS lacks the relational capability. NoSQL (Not Only SQL) was developed as a solution addressing both limitations. NoSQL is a big data management system with capabilities of SQL and horizontal scaling that enables replicating and partitioning of data over numerous servers (Cattell, 2011). NoSQL has often shown high computational performance (Li & Manoharan, 2013). Geospatial data pose additional challenge in that they also must store the spatial, and in some cases, temporal dimensions of the data. The spatial dimension can be the simple locational attributes and topology. Although locational attributes can be implemented relatively easily, topology is more challenging. Due to these additional attributes, spatial capabilities are often added as extensions even in RDBMS such as the POSTGIS of the PostgreSQL, the spatial Hadoop, etc.

A hierarchical and programmable data storage system: NetCDF

There are also other formats of data storage that are well suited to big data, often called programming model. For example, multidimensional array-based data such as images can be stored efficiently using the Hierarchical Data Formats (HDF). Widely used specifications of this format are the netCDF (Network Common Data Format) and the HDF5 formats which are efficient for storing, accessing, and computing array data of different types. The NetCDF stores the data and the metadata in multiple dimensions in just one file. The contents of the file are divided into dimensions, variables and attributes (Li et al., 2003; Rew & Davis, 1990). It is also extensible as more data and metadata can be added at any time later using the programming possibilities realized through APIs. Additionally, the files stored can be chunked into blocks of favourable sizes for subsequent distributed processing.

When storing a data in a NetCDF format, the dimensions, variables, and attributes of the data are defined. Dimension describes the extents of the data coverage spatially, temporally, and thematically. Variables are values of the thematic data and the dimensions. Attributes can be anything that describes the file in general (global attributes) or a dimension or a variable or a subgroup of the file. As figure 4 below depicts, NetCDF files are self-describing files with both data and metadata stored in the same file.

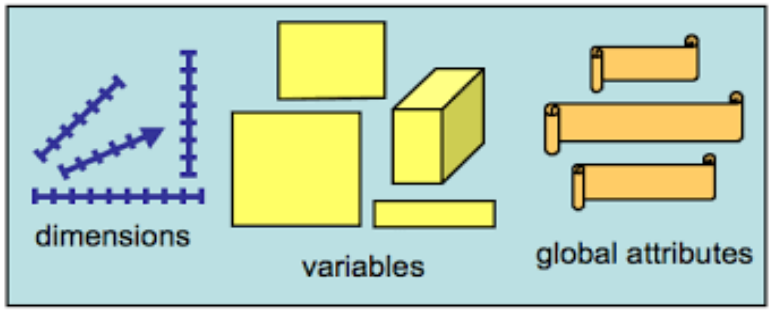

Figure 4. The NETCDF file model[4]

Testing of NetCDF in NIBIO

The python API of the NetCDF file specification is installed and tested in an Anaconda development environment. A NetCDF file is first operated with the NETCDF-4 format which is the newest version with HDF5 like standard. It enables to implement all the functionalities of the HDF5.

Dimensions of space, time, and feature (data) levels are defined first. One can either define the size of each dimension or leave it as limitless (only in NETCDF-4). The northing and easting are defined as the space dimension of the images to be saved. The time and level dimensions are left to be limitless. This implies that we have defined the horizontal and vertical sizes of the geospatial data we are going to save but we can save data of numerous bands over multiple limitless times.

NETCDF-4 is hierarchical. This means it offers possibilities of groups and subgroups in a similar fashion as directories and subdirectories in the same file, providing opportunities to store data of limitless size. Sentinel-2 images of one tile acquired over one year are stored in a single NetCDF file as a test of the solution. Thus, it stores satellite image time series (SITS) data. The NetCDF file is ready for multiple access and distributed processing by using different chunk sizes. The metadata is also explorable without opening the entire file. For example, Sentinel-2 images taken over the land area of Norway in the year 2019 is stored in NetCDF format. Images of each Sentinel-2 tile are stored as one NetCDF file including the dimensions of northings, eastings, spectral bands, and time, resulting in 4D array. This stored data

[4] https://www.unidata.ucar.edu/software/netcdf/workshops/2011/datamodels/NcFile.html

has shown to be very effective in time series analysis. Such data format organised in multidimensional arrays amount to what is called datacubes, discussed below in more detail.

Datacubes

A datacube is a multi-dimensional ("n-D") array of values. Typically, the term datacube is applied in contexts where these arrays are massively larger than the hosting computer's main memory; examples include multi-terabyte/petabyte data warehouses and time series of image data. The datacube is used to represent data (sometimes called facts) along some measure of interest. In satellite image timeseries these measures would be Latitude and Longitude coordinates and time; a fact would be a pixel at a given space/time coordinate as taken by the satellite (following some processing that is not of concern here). Even though it is called a cube (and the examples provided above happen to be 3-dimensional for brevity), a data cube generally is a multi-dimensional concept which can be 1-dimensional, 2-dimensional, 3-dimensional, or higher-dimensional. In any case, every dimension represents a separate measure whereas the cells in the cube represent the facts of interest. Sometimes cubes hold only few values with the rest being empty, i.e.: undefined, sometimes most or all cube coordinates hold a cell value. In the first case such data are called sparse, in the second case they are called dense, although there is no hard delineation between both.

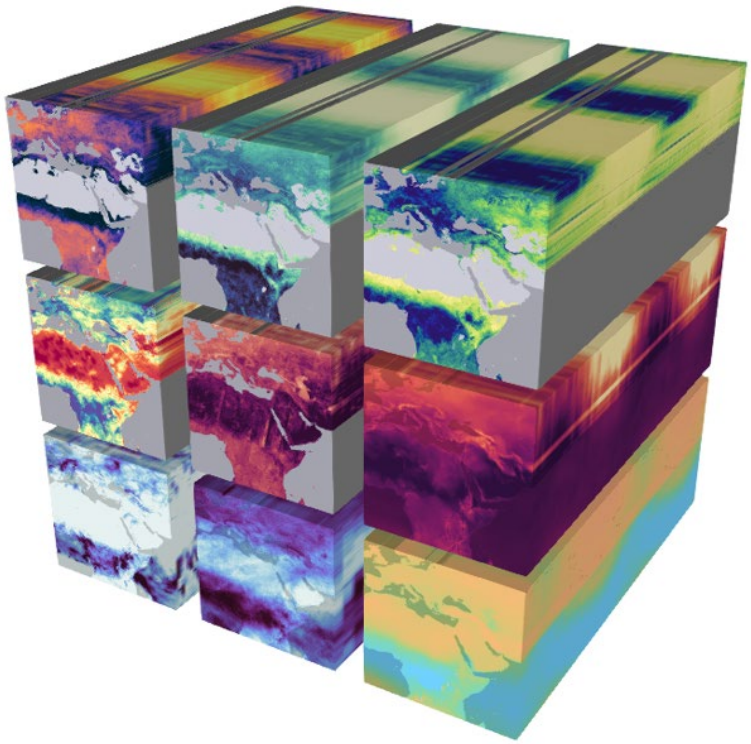

Figure 5. A representation of a datacube from Copernicus (Mahecha et al., 2020).

There are three main datacube models that are currently available for the scientific community. In addition, several organizations such as ESA are using datacube infrastructures to distribute data.

Rasdaman

The most advanced and powerful datacube is Rasdaman[5] ("raster data manager"). It is an Array DBMS which adds capabilities for storage and retrieval of massive multi-dimensional arrays, such as sensor, image, simulation, and statistics data. A frequently used synonym to arrays is raster data, such as in 2-D

[5] http://www.rasdaman.org/

raster graphics; this has motivated the name Rasdaman. However, Rasdaman has no limitation in the number of dimensions - it can serve, for example, 1-D measurement data, 2-D satellite imagery, 3-D x/y/t image time series and x/y/z exploration data, 4-D ocean and climate data, and even beyond spatio-temporal dimensions.

Open Datacube

The Open Data Cube[6] (ODC) is an Open Source Geospatial Data Management and Analysis Software project that helps harnessing the power of satellite data. At its core, the ODC is a set of Python libraries and PostgreSQL database that helps working with geospatial raster data.

The ODC seeks to increase the value and impact of global Earth observation satellite data by providing an open and freely accessible exploitation architecture. The ODC project seeks also to foster a community to develop, sustain, and grow the technology and the breadth and depth of its applications for societal benefit.

Earth System Data Lab

The Earth System Data Lab[7] (ESDL) is not only an empty infrastructure, but is meant to offer a service to Earth System scientists: the provision of a versatile and simple-to-use online laboratory, i.e. infrastructure and tools, to analyse the various data sets in the cube. The ESDL provides access to a series of highly curated "data cubes"'. These analysis ready data were collected after previous user consultations and pre-processed to common spatio-temporal resolutions ready for computations in a cloud (or cloud-based computations). Most of them have been derived from Earth Observation, but the compilation also includes model or re-analysis data if deemed useful. Further dimensions can be added as a result of an analysis. With this at hand, users may tackle a great variety of questions and come up with innovative applications.

4.2 A comprehensive framework: Hadoop

A brief description of Apache Hadoop was given in chapter 4.1 above. Apache Hadoop[8] is a collection of open-source software utilities that facilitate using a network of many computers to solve problems involving massive amounts of data and computation. It provides a software framework for distributed storage and processing of big data using the MapReduce programming model. Originally designed for computer clusters built from commodity hardware — still the common use — it has also found use on clusters of higher-end hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common occurrences and should be automatically handled by the framework. The base Apache Hadoop framework is composed of the following modules[9]:

- **Hadoop Common:** contains libraries and utilities needed by other Hadoop modules;
- **Hadoop Distributed File System (HDFS):** a distributed filesystem that stores data on commodity machines, providing very high aggregate bandwidth across the cluster;
- **Hadoop YARN:** a platform responsible for managing computing resources in clusters and using them for scheduling users' applications;
- **Hadoop MapReduce:** an implementation of the MapReduce programming model for large-scale data processing.

[6] https://www.opendatacube.org/

[7] https://www.earthsystemdatalab.net/

[8] https://hadoop.apache.org/

[9] The term Hadoop is often used for both base modules and sub-modules and also the ecosystem, or collection of additional software packages that can be installed on top of or alongside Hadoop, such as Apache Pig, Apache Hive, Apache HBase, Apache Phoenix, Apache Spark, Apache ZooKeeper, Cloudera Impala, Apache Flume, Apache Sqoop, Apache Oozie, and Apache Storm.[13]

The core of Apache Hadoop consists of a storage part, known as Hadoop Distributed File System (HDFS), and a processing part which is a MapReduce programming model. Hadoop splits files into large blocks and distributes them across nodes in a cluster. It then transfers packaged code into nodes to process the data in parallel. This approach takes advantage of data locality, where each node manipulates the data it has access to. This allows the dataset to be processed faster and more efficiently than it would be in a more conventional supercomputer architecture that relies on a parallel file system where computation and data are distributed via high-speed networking.

As a follow up to the paper on distributed file system, Google Inc. also published a paper that outlined the framework of efficient and distributed data processing (Dean & Ghemawat, 2008). This paper was the basis of MapReduce, a highly scalable and reliable programming model for parallel and distributed computation which involves mapping and reducing steps (Dean & Ghemawat, 2004). MapReduce is a particularly relevant element in the Hadoop framework. It uses a divide and conquer approach where the tasks are divided into smaller tasks and into mapper and reducer tasks. It is implemented over the distributed file systems such as the HDFS. In the framework, the mapper performs a set of tasks by mapping input key/value pairs to a set of intermediate key/value pairs. The intermediate results can also be further mapped. Then, the reducer reduces a set of intermediate values into smaller number of values, which may further be reduced to even smaller set of values. Additional frameworks such as shuffle and sort are also available to organize the inputs, the intermediate results, and the clusters to facilitate the process.

MapReduce is applicable to tasks that can be separated into a map and reduce framework. It lacks the flexibility to handle other sets of tasks. The user is expected to find out if the task can be mapped and reduced and then program it in such framework. The greatest limitation is this lack of flexibility which inhibits application to iteration, interactive tasks, etc.

The Hadoop infrastructure can be useful for processing raster dataset such as satellite images or climate or other environmental models. The framework could be implemented, after thorough evaluation of the need, by the geomatics department in cooperation with the IT department, making it available for everyone needing it within NIBIO and organize training activities.

4.3 Distributed computing: the Dask framework

Dask[10] is an open-source framework that allows developers to build their software in coordination with scikit-learn, pandas, and NumPy packaged of the Python programming language. Further, Dask deploys distributed processing using common Python terminologies, making it easy to implement (Rocklin, 2015). It is a very versatile tool that works with a wide array of workloads. Dask includes two parts: i) a task scheduling component for building dependency graphs and scheduling tasks; ii) distributed data structures with APIs similar to Pandas Dataframes or NumPy arrays. Dask has a variety of use cases and can be run with a single node and scale to thousand node clusters.

Dask is relatively straightforward to implement and scale up. Furthermore, it does not require many changes to the programming framework as it schedules the tasks using common python modules. In the framework, one computer serves as a task scheduler and other computers can be connected to the scheduler as workers. It can theoretically include an unlimited number of workers in Linux; however, it is limited to about 1000 machines in Windows (which is already a fairly large number).

When a scheduler and several workers are setup, the RAM accessible for the client is the total Sum of all the RAMs of the connected workers and the scheduler (if the scheduler is also connected as worker). A client first connects to the distributed system by connecting to the scheduler and sending a job. The scheduler successively distributes the job to the workers in the system by splitting the data into chunks.

[10] https://dask.org/

Once the scheduled activities are executed in a worker, output data are returned to the scheduler who will merge them before returning the result to the client.

The major advantage of the system is that, large files that would otherwise not be processed by a single computer, due to inadequate RAM, can be opened and processed. Additionally, for processes that take a long time, the distributed processing using several machines shortens the processing time. However, there is a cost of file transaction between the processing nodes (workers). Some work around is available to reduce this file transmission costs. While one work around is scattering the data prior to processing, another work around is implementing Dask on top of distributed file system such as the HDFS.

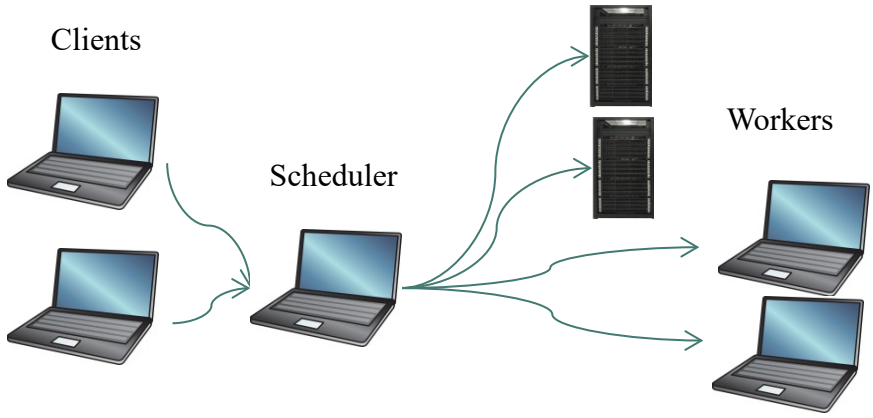

Figure 6. The Dask cluster architecture.

Dask in NIBIO in combination with Xarray and NetCDF

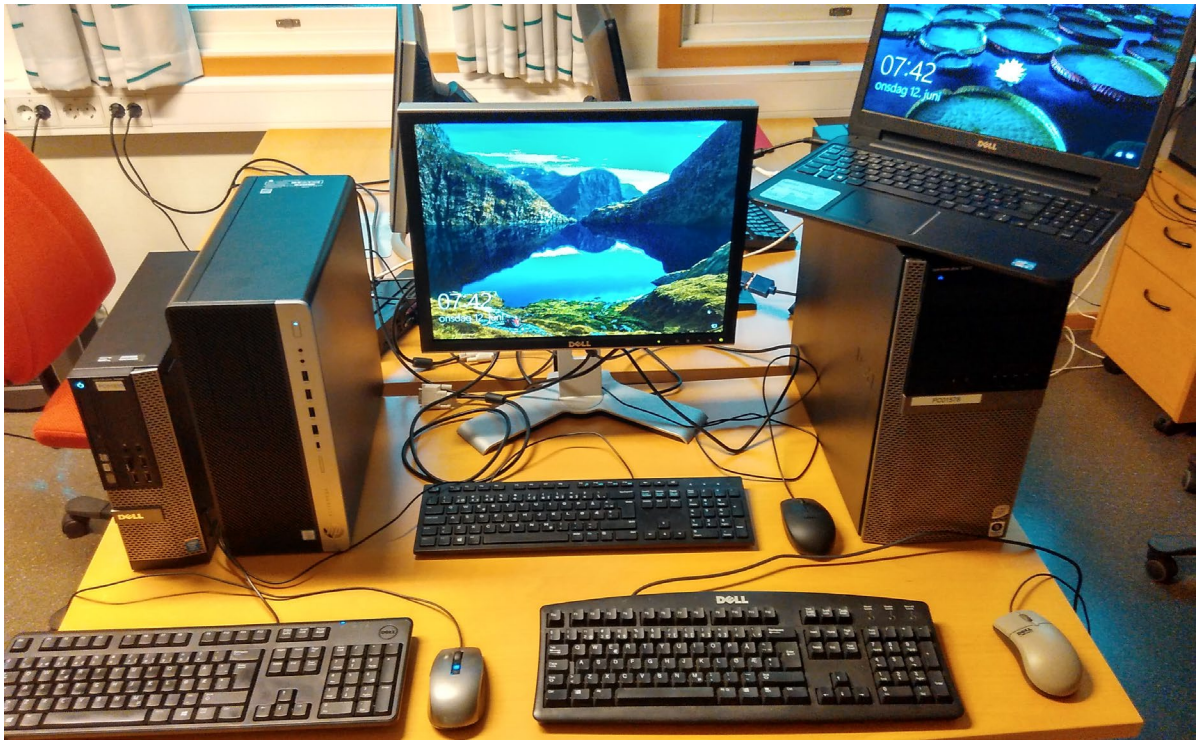

Figure 7. Some of the machines used for testing Dask at NIBIO (photo by Jonathan Rizzi)

We tested the framework using personal computers interconnected following the guidelines given in the Dask manual. The distributed cluster is composed of three computers with 64 GB total RAM. One PC served as scheduler and the others as workers. A client is then connected to the cluster using the Python API of Dask. After connecting the client, it is theoretically possible to run any python function in the cluster provided that the data used is either located at a storage medium accessible by the workers or on all workers or distributed to the computers in the process.

To test these functionalities, two types of medium sized data are used. First, a NetCDF file with sentinel-2 images of size 8 GB was used to compute NDVI. Opening this in python NumPy is not successful using normal computers with 8 GB RAM. The goal here is to compute the NDVI of the image and store it using the NetCDF data specification.

A python script is then written to open the file and compute the NDVI so that it is computed using the cluster and then save the computed NDVI to the NetCDF file specified. The script makes use of the Xarray python module built on top of Dask. Xarray is a python-based package for opening and processing multi-dimensional arrays (Hoyer & Hamman, 2017). Its major advantages are: firstly, it merges the labelled data format of dataframes that are often used for statistical analysis and the pure array format of NumPy which is often used for image processing creating labelled multidimensional arrays. Secondly, it leverages the possibility of accessing data out of memory. Both advantages are important in data science to implement statistical analysis directly on multi-dimensional arrays and to process files that are larger than the memory of the machines. Xarray's advantages are even further utilized when combined with Dask as it offers distributed processing. Large files can thus be chunked into pieces and processed by different processers and/or cores.

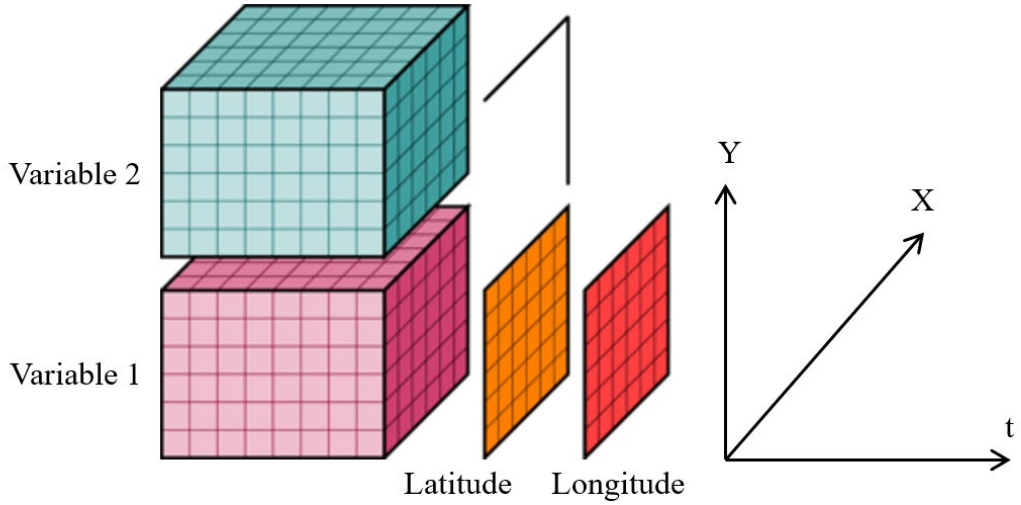

Figure 8. The Xarray (and Dask array) data model.

The functionalities of Xarray and Dask are utilized even better when combined with file storage systems that are ready for distributed processing. The hierarchical data formats with data chunked in to blocks such as HDF5 and NetCDF and the distributed data formats such as the HDFS are suitable. Dask is in fact capable of working with both HDF5/NETCDF and HDFS files. An inspiring successful integration of Dask, Xarray and NetCDF is implemented by the geoscience community called EarthCube in the PANGEO project (Ryan et al., 2017).

The approach leveraged two advantages: firstly, the file which could not have been processed by a single computer, is now processed using the Xarray out of memory capability combined with the distributed system. Secondly, the speed of computation is enhanced as follows based on the alternative use of the three computers (A=1 processor, 4 cores, 8 GB RAM, 2.3 GHZ, B=1 processor, 8 cores, 8 GB RAM, 3.4 GHZ, C=1 processor, 8 cores, 32 GB RAM, 3.4 GHZ).

Table 2. Description of the different combinations of machines used in the test.

Method	Cluster	Performance (min)	Comment
Pure python NumPy	A	Crushed	RAM exceeded
Single computer Xarray	A	13.6	Serial
Dask local cluster	A	3.4	4 cores in parallel
Dask distributed cluster	A + B	4.13	Costly data transmission
Dask distributed cluster	A + C	1.7	Benefits of large RAM from C
Dask distributed cluster	A + B + C	1.6	Not much improvement

Based on this we implemented a real task of creating cloudless and shadow free Sentinel-2 image mosaic over the entire land area of Norway. The task requires implementation of the following tasks:

- Define the month
- Define the tile
- Search for images of low cloud percentage using the API for the
- Open the files using Xarray and chunks
- Stack the images of the same month in to 4 -dimensional array stack or tensor.
- Then reduce the data using suitable algorithm, median or quantile are implemented.
- This is a heavy task even for one tile
- This is a 4D data cube

It would have been impossible to open this big data by a personal PC if it was using normal file opening in python or any other program. It is the chunking and delayed processing offered by Xarray and Dask that made the processing possible.

A more robust and systematic test of Dask together with Xarray and NetCDF was performed, and a technical paper has been written and is currently under review.

4.4 A framework for clusters of computers: Apache Spark

Apache Spark[11] is an open-source distributed general-purpose cluster-computing framework. Spark provides an interface for programming entire clusters with implicit data parallelism and fault tolerance. Spark extends the distributed processing framework of MapReduce and deals with most of its drawbacks. Further, Spark has APIs in wide range of programming languages.

Apache Spark requires a cluster manager and a distributed storage system. For cluster management, Spark supports the native Spark cluster, Hadoop YARN, Apache Mesos or Kubernetes. For distributed storage, Spark can interface with a wide variety, including Alluxio, HDFS, MapR File System (MapR-FS), Cassandra, OpenStack Swift, Amazon S3, Kudu, Lustre file system. Spark also supports a pseudo-distributed local mode, usually used only for development or testing purposes, where distributed storage is not required and the local file system can be used instead; in such a scenario, Spark is run on a single machine with one executor per CPU core.

Apache Spark include several components:

- **Spark Core:** it is the foundation of the overall system. It provides distributed task dispatching, scheduling, and basic I/O functionalities, exposed through an application programming interface (for Java, Python, Scala, and R) centred on the RDD abstraction.

[11] https://spark.apache.org/

- **Spark SQL:** it is a component on top of Spark Core introducing a data abstraction called DataFrames, which provides support for structured and semi-structured data. Spark SQL provides a domain-specific language (DSL) to manipulate DataFrames in Python, Java, or Scala. It also provides SQL language support, with command-line interfaces and ODBC/JDBC server.
- **Spark Streaming:** it uses Spark Core's fast scheduling capability to perform streaming analytics. It ingests data in mini-batches and performs RDD transformations on those mini-batches of data. This design enables the same set of application code written for batch analytics to be used in streaming analytics, thus facilitating easy implementation of lambda architecture[12].
- **MLlib Machine Learning Library:** it is a distributed machine-learning framework on top of Spark Core that, due in large part to the distributed memory-based Spark architecture, is much as faster than other disk-based implementations.
- **GraphX:** it is a distributed graph-processing framework on top of Apache Spark.

[12] A lambda architecture is a data-processing architecture designed to handle massive quantities of data by taking advantage of both batch and stream-processing methods. This approach to architecture attempts to balance latency, throughput, and fault-tolerance by using batch processing to provide comprehensive and accurate views of batch data, while simultaneously using real-time stream processing to provide views of online data. The two view outputs may be joined before presentation.

5 Large infrastructures for big data

In recent years there is an increasing number of solutions for storing and processing big data through the cloud on very large, shared infrastructures. These solutions can be useful when the cost of an in-house solution is too large or when there are insufficient human resources to maintain these infrastructures. The platforms usually have a cost associated with their use, which is not fixed as traditional software licenses, but is variable depending on the amount of time spent on the platform, on the number of transaction or on other variable parameters. Costs change through time and is not presented in this document, but associated costs should also be considered when planning for the use of one of these platforms.

5.1 Uninett Sigma2

UNINETT Sigma2 AS[13] is the national e-infrastructure for large-scale data and computational science in Norway. Sigma2 provide services for high-performance computing and data storage to individuals and groups involved in research and education at all Norwegian universities and colleges, and other publicly funded organizations and projects.

The infrastructure has three main components:

- **High Performance Computing:** the High Performance Computing (HPC) service provides access to computing facilities and software with a far greater capacity than is normally available at department and faculty levels.
- **Data Storage:** the service offers data storage facilities to researchers who require a platform to store, share and manage large, active scientific datasets.
- **NIRD Toolkit:** the NIRD Toolkit is a Kubernetes based cloud infrastructure, similar to Google, Azure or Amazon Kubernetes Clouds (these three will be better described in the following sections).

UNINETT Sigma2 also provides other services, which can be particularly useful for institutes such as NIBIO since they are prepared for researchers as final uses, while other platforms are more tailored to the needs of private companies.

The additional services of Sigma2 are the following:

- **Advanced User Support:** Advanced User Support offers help that goes beyond the general (basic) user support to provide specialised and more in-depth competence to a research group or community.
- **CRaaS - Course Resources as a Service:** CRaaS is a service for researchers who require e-infrastructure resources to be used in a course or a workshop for research purposes.
- **Data Planning:** EasyDMP is a service that offers researchers with minimal experience in data management, a simple way of creating Data Management Plan (DMP).
- **Research Data Archive: t**he Research Data Archive is a repository of valuable research data. The Research Data Archive provides users opportunities to archive, publish and share their data openly.
- **Sensitive Data Services:** the Sensitive Data Service (TSD) provides a platform to store, compute and analyse research sensitive data in compliance with Norwegian regulations regarding individuals' privacy.

[13] https://www.sigma2.no/

5.1.1 Sigma2 advanced services – The NIRD toolkit

The NIRD toolkit represent an advanced and powerful set of tools that can be used in connection with the Sigma2 storage. The available software runs in containers to ensure high portability of the tools and reproducibility of the results. It is highly customizable, meaning that it is possible to have the desired tools in different versions. Further, the NIRD toolkit allows pre/post processing analysis, data intensive processing, visualization, artificial intelligence, and machine learning platform.

The service available within the NIRD toolkit are the following:

- Spark
- RStudio
- Jupyter
- JupyterHub
- MinIO
- Deep Learning Tools that includes preinstalled libraries such as PyTorch, Tenserflow, Keras, CNTK, mxnet, Theano and caffe2

5.1.2 A test of the Sigma2 infrastructure within NIBIO

To test the service, we defined a small project aiming to create a cloud free image mosaic over the whole of Norway during May 2018. Although we have not explored different ways of improving computational speed, the process itself created the image mosaic without crashing.

The used dataset

The dataset used in this test includes all Sentinel-2 satellite images acquired over the entire land of Norway in May 2018. Norway is covered by 68 Sentinel-2 tiles. Each tile has several acquisitions during a month. The images are filtered based on the cloud coverage and atmospheric correction. Images with cloud coverage less than 70% and that are atmospherically corrected to level 2A are chosen and downloaded. This results in a number between 6 and 10 images for each tile.

Method

The procedure used to identify cloud free images is to use the computed median, min and percentile values obtained for each pixel and each band over the image datasets (using only the images previously selected).

Since there are 68 Sentinel-2 tiles covering Norway, the procedure needs to be repeated 68 times. This approach is perfectly suitable for parallel computation. For such problems Sigma2 offers array computation where it is possible to parallelize the task and distribute the job over the HPC. The process involves downloading the files from the ESA image server, that unfortunately allows only two parallel accesses for one user, representing a strong limitation (and bottleneck) for the parallel processing. The consequence is that it is possible to process only two tiles at a time. Therefore, this is the number of parallel processes that are executed. One can also program in such a way that the tiles are chunked into blocks and computation is parallelized over the blocks.

Processing one tile by itself is computationally memory demanding. One normal PC cannot perform the task in a serial computation. It takes long time even when it is divided into pieces and serialized. Therefore, considering that the computation involves tensors (3D arrays), a framework of multiprocessing is implemented using the Dask framework. Dask_jobqueque is used to manage the parallel job of multiprocessing a single tile over multiple cores. Dask_jobqueque is a Dask framework for distributed processing on HPC's.

5.2 Amazon Web Services

Amazon Web Services[14] (AWS) is a subsidiary of amazon.com, which provides an on-demand Cloud Computing platform to individuals, companies, and governments on a paid-subscription basis. Amazon Web Services is the oldest and the most experienced player in the cloud market. As one of the oldest cloud providers, it has established a bigger user base, as well as bigger trust and reliability factors.

5.2.1 AWS Storage Services

This is one area where AWS does research into offering a hybrid platform through its Storage Gateway. Gateway offers a secondary archival storage option in conjunction with Amazon's sole backup feature, Glacier. Users can opt for simple object storage with S3 or block storage for large containers with their elastic block feature; this one operates in conjunction with E2B. In addition, the elastic file storage expands your capability as you create files, which is ideal for large corporations that generate a lot of data.

Amazon Web Services also provides several SQL-supported databases, an ElastiCache feature to provide additional memory, and a data migration service.

- **Storage:** Simple Storage Service (S3), Elastic Block Storage (EBS), Elastic File System (EFS), Storage Gateway, Snowball, Snowball Edge, Snowmobile
- **Database:** Aurora, RDS, DynamoDB, ElastiCache, Redshift, Neptune, Database migration service
- **Backup Services:** Glacier

5.2.2 AWS Compute Features

The primary computing service is the Amazon Elastic Compute Cloud (E2C). The E2C integrates with most Amazon Web Services, promoting compatibility and a high degree of flexibility, which allows database administrators to optimize for cost. The scalable cloud platform allows you to scale up or down in minutes, and it can deploy thousands of server instances at lightning speed.

Using the AWS auto scaling monitor puts machine learning to use by monitoring apps and scaling to capacity according to the current requirements. Amazon guarantees 99.99 percent of availability of the AWS.

Another important compute element of AWS is the Amazon Elastic Container Service (ECS): This scalable container orchestration supports Docker containers through a series of API calls. With this ability, you can begin or end Docker-enabled apps, query the state of an application, manage website IP address blocking and unblocking, and access security groups, IAM roles, CloudWatch events, CloudTrail logs, and CloudFormation templates. There is also an ECS registry feature and a container service for Kubernetes.

Other AWS Compute features include: AWS Beanstalk, Amazon Lightsail, AWS Serverless Application Repository, VMware Cloud for AWS, AWS Batch, AWS Fargate, AWS Lambda, AWS Outposts, Elastic Load Balancing

[14] https://aws.amazon.com

5.3 Microsoft Azure

Microsoft Azure[15] was launched in 2010 with the intent to provide a competent Cloud Computing platform for businesses. Since its inception, Microsoft Azure has shown a great progress among its competitors.

5.3.1 Azure Storage Services

Azure offers a dedicated storage option called Blob Storage. This is reserved for unstructured, REST-based object warehousing. Like AWS, it also has solutions for large-scale data storage and high-volume, critical workloads with their Queue Storage and Data Lake Store. This platform also provides users with the largest array of databases, which support three different SQL-based formats, and its Data Warehouse gives room to grow.

The support that Azure provides for SQL isn't limited to storage. Its Server Stretch database is a hybrid that offers on- and off-premises storage for companies that use Microsoft SQL Server for their enterprise but might utilize other protocols on the cloud. This is the only company of the three (the others are Amazon and Google) that has a backup recovery system, which is in addition to its archival and standard system backups.

- **Storage:** Blob Storage, Queue Storage, File Storage, Disk Storage, Data Lake Storage
- **Database:** SQL database, Database for MySQL, Database for PostgreSQL, Data warehouse, Server Stretch database, Cosmos DB, Table storage, Redis cache, Data Factory
- **Backup Services:** Archival storage, Recovery backups, Site recovery

5.3.2 Azure Compute Features

Azure compute features rely on a network of virtual machines to enable a range of computing solutions that include development, testing, data centre extensions, and app deployment. It's based on an open-source platform compatible with Linux, Windows servers, SQL Server, Oracle, and SAP. Azure also offers a hybrid model that combines on-premises and public clouds, and it can be integrated into global load balancing.

Azure Kubernetes Service (AKS) is a serverless container system that allows containerized applications to be deployed and managed faster. It offers a seamless continuous integration/continuous delivery (CI/CD) experience, security, and enterprise governance to unite diverse teams working within a virtual office setting on a single platform.

Other Azure compute features include: Platform-as-a-service (PaaS), Function-as-a-service (FaaS), Service Fabric, Azure Batch

5.4 Google Cloud Platform

Google Cloud Platform[16] (GCP), which is offered by Google, is a suite of Cloud Computing services that runs on the same infrastructure that Google uses internally for its end-user products such as Google Search engine, YouTube, and more.

5.4.1 Google Storage Services

Google Cloud Platform (GCP) offers basic storage and database support, but little else. Its storage solutions are similar to what GCP provides customers in the computing department and provide both

[15] https://azure.microsoft.com

[16] https://cloud.google.com/

SQL and NoSQL database support. GCP has a transfer appliance that's similar to AWS Snowball, and several online transfer services are available.

- **Storage:** Cloud storage, Persistent disk, Transfer appliance, Transfer service
- **Database:** Cloud SQL, Cloud Bigtable, Cloud Spanner, Cloud Datastore
- **Backup Services:** Nearline (frequently accessed data), Coldline (infrequently accessed data)

5.4.2 Google Cloud Compute Features

The Google Cloud is well integrated with Kubernetes containers (the company behind Kubernetes had a hand in developing the GCP platform, and it's their main service model). Further, Google Cloud also supports Docker containers.

Cloud Functions is still in the beta phase, but it shows a lot of promise with various features. It is possible to allow the service to manage resources and deploy apps, automatically scale according to traffic, or use in real-time, and deploy code from Google Cloud, Firebase, or Assistant. It is also possible to call functions from any network or device using HTML.

Other GCP compute functions include: Google App Engine, Docker Container registry, Instant groups, Compute Engine, Graphic processing unit (GPU), Knative

5.4.3 Google Earth Engine

Google Earth Engine[17] (GEE) is a web-based tool which is not part of the GCP but can be used in connection with the Google Cloud storage. GEE combines a multi-petabyte catalogue of satellite imagery and geospatial datasets with global-scale analysis capabilities and is a platform for scientific analysis and visualization of geospatial datasets.

Google Earth Engine hosts satellite imageries and stores them in a public data archive that includes historical earth images going back more than forty years. The images, ingested daily, are then made available for global-scale data mining.

Google Earth Engine also provides APIs and other tools to enable the analysis of large datasets. Although java script is its native scripting language, its python API is powerful as it can be combined with other python classes and functions.

GEE is a highly useful tool for NIBIO, specially for international projects that operate in countries where more detailed national data do not exist. Additionally, the computational infrastructure is highly useful for countries that lack well developed IT infrastructure. NIBIO's expanding international cooperation can benefit from using GEE.

[17] https://earthengine.google.com/

6 Methods for geospatial big data analysis

When it comes to the analytical part of big data, the methods depend upon the purpose of the analysis and the type of the data. However, some analytical methods are also developed together with big data, as they are well suited for this purpose. Traditional statistical and other data analysis methods are still relevant. However, Fan et al. (2014) warn that caution must be taken about the salient features of big data that makes traditional statistics invalid in some cases.

The purposes of analyses can be one of descriptive, diagnostic, predictive or prescriptive. Analyses that are conducted to understand what happened, or describe processes and phenomena, using descriptive statistics, data clustering, classification, etc. are referred to as descriptive analysis.

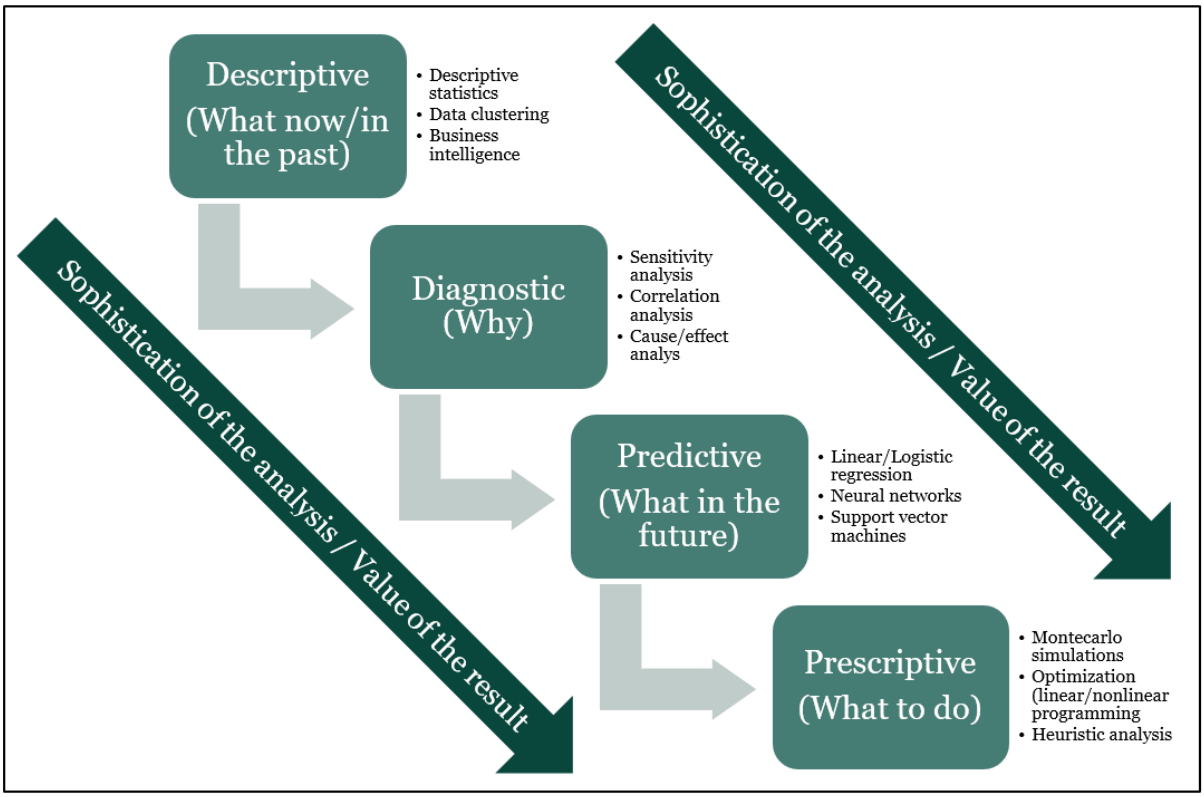

Figure 9. Different types of geospatial analysis.

The unique features of big data that are important for the analysis step are the large volumes of data, the high dimensionality of the data and the heterogeneity of the data in terms of data format (e.g. raster, vector) and data type (integer, character, floating).

As mentioned earlier, Artificial intelligence include machine learning methods, which are particularly relevant for spatial big data analysis. An advancement and specialization of machine learning is represented by deep learning. In the following sections several machine learning and deep learning techniques used for spatial big data will be briefly presented. This white paper does not aim at providing a detailed description of each method, which can be found in scientific literature cited later in the document or in many examples of applications and implementations easily discoverable on the internet. Further, there is a constant development of libraries implementing these algorithms. Therefore, a detailed description of libraries that implemented those algorithms would immediately be outdated and is not included in this report.

6.1 Machine Learning

Machine learning is a subfield of artificial intelligence focusing on the definition and implementation of algorithms that can learn through experience and data without being explicitly programmed. Machine learning algorithms build a mathematical model based on sample data (training data) that allow predictions to be made without being explicitly programmed to do so. Machine learning is based on the use of statistical models and several methods can belong to both the world of statistics and machine learning. Several authors claim that statistics and machine learning can use the same method for different purposes (e.g. linear regression), and there has been a large debate that is still ongoing about the boundary between these two disciplines. This is of no further concern here, and all the methods presented hereafter can be considered within the world of the machine learning (with several also commonly used in statistics).

Regression

A set of statistical processes for estimating the relationships between a dependent variable (often called the 'outcome variable') and one or more independent variables (often called 'predictors', 'covariates', or 'features'). The most common form of regression analysis is linear regression, in which a researcher finds the line (or a more complex linear combination) that most closely fits the data according to a specific mathematical criterion. There are several types of regression such as Linear regression; Simple linear regression; Logistic regression; Nonlinear regression and several others.

Classification, clustering (KNN, Kmeans)

It is considered an instance of supervised learning, i.e., learning where a training set of correctly identified observations is available. Often, the individual observations are analysed with respect to a set of quantifiable properties, known variously as explanatory variables or features. These properties may variously be categorical, ordinal, integer-valued or real-valued. Other classifiers work by comparing observations to previous observations by means of a similarity or distance function. An algorithm that implements classification is called classifier.

The task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields. Clustering is not one specific algorithm by itself, but it is the general task to be solved. It can be achieved by various algorithms or models, such as Connectivity models; Centroid models; Distribution models; Density models and several others.

Dimensionality reduction (Principal Component Analysis)

The transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful properties of the original data, ideally close to its intrinsic dimension. Working in high-dimensional spaces can be undesirable for many reasons; raw data are often sparse because of the curse of dimensionality, and analysing the data is usually computationally intractable. Dimensionality reduction is common in fields that deal with large numbers of observations and/or large numbers of variables. Methods are commonly divided into linear and non-linear approaches. Approaches can also be divided into feature selection and feature extraction. Dimensionality reduction can be used for noise reduction, data visualization, cluster analysis, or as an intermediate step to facilitate other analyses.

Naive Bayes

It is a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naïve) independence assumptions between the features. They are among the simplest Bayesian network models. But they could be coupled with Kernel density estimation and achieve higher accuracy levels. Naïve Bayes classifiers are highly scalable, requiring several parameters linear in the number of

variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

Support Vector Machine

A support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like detection of outliers. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin, the lower the generalization error of the classifier. Although it is a well performing method, support vector machine has limitation in its applicability due to the high computational needs.

Decision Tree (Random Forest)

It is a predictive modelling approach that uses a decision tree as a predictive model to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. Decision trees are among the most popular machine learning algorithms given their intelligibility and simplicity. A classification process that involves ensemble of decision trees with multiple number of trees is called Random Forest.

6.2 Deep learning

Deep learning is the subset of machine learning methods that is based on artificial neural networks with representation learning. Deep learning uses multiple layers to progressively extract higher level features from the raw input (e.g., in remote sensing image processing, lower layers may identify edges of objects identified on the ground, while higher layers may identify the concepts relevant to a human such as buildings). Learning can be supervised, semi-supervised or unsupervised. Examples of architectures used in deep learning are:

Deep neural network

It is a neural network with multiple layers between the input and output layers. Deep neural networks find the correct mathematical manipulation to turn the input into the output, whether it be a linear relationship or a non-linear relationship. The network moves through the layers calculating the probability of each output. Each mathematical manipulation as such is considered a layer, and complex deep neural networks have many layers, hence the name "deep" networks.

Deep belief network

It is a class of deep neural networks composed of multiple layers of latent variables ("hidden units"), with connections between the layers but not between units within each layer. When trained on a set of examples without supervision, a deep belief network can learn to probabilistically reconstruct its inputs. The layers act as feature detectors and after the learning step, the network can be further trained with supervision to perform classification.

Recurrent neural network

It is a class of neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behaviour. Recurrent neural networks can use their internal state (memory) to process variable length sequences of inputs.

Convolutional neural network

It is the most commonly class of deep neural networks that is applied to analyse images. Convolutional neural networks are also known as shift (or space) invariant artificial neural networks, based on their shared-weights architecture and translation invariance characteristics. A convolutional neural network is a regularized version of a multilayer perceptron. The typical problem of overfitting found in multilayers perceptron is addressed by taking advantage of the hierarchical patten in data and assemble more complex patterns using smaller and simpler patterns. Therefore, convolutional neural networks considered networks with low connectedness and complexity.

6.3 Data mining

Data mining refers to the process of knowledge discovery in big data. It involves extraction of useful information through discovering trends, patterns, and correlations in the dataset. Data mining is sometimes referred to also by other terms such as knowledge extraction, information discovery, information harvesting, etc. Methods from various disciplines are involved in data mining, including traditional statistical data analysis. Data mining also overlaps with machine learning in many of the used tools, as it involves building relationships and models. It also involves patterns and anomaly detection. The fundamental difference between data mining and machine learning is that data mining primarily concerns with describing data and extracting patterns and relationships, while machine learning goes beyond these and does predictions and generate new data. Additionally, while data mining relies on the entire dataset to learn the patterns, machine learning can learn from a subset of the dataset. The purpose of data mining is to help focus on the valuable data in an analysed database, representing a type of data reduction. Finally, data mining also helps discovering previously unknow patterns and relationships in the data.

7 Opportunities and challenges related to spatial big data at NIBIO

There is considerable potential from the use of spatial big data at NIBIO. Part of this potential is already explored. Many researchers have competences and are already working with spatial big data in different department, particularly in the centres for precision agriculture and precision forestry. However, most of the work is still done by independent researchers or very small group of researchers with limited coordination between the groups. This sometimes can represent a limitation. The full potential requires a stronger cooperation between different departments. Sharing information about data, methods and available tools can give the possibility of further improvements in many sectors.

This chapter is based on a SWOT (Strengths, Weaknesses, Opportunities, and Threats) analysis, an analysis of the risk, criticalities and bottlenecks in spatial big data and some ideas for possible future developments of spatial big data within NIBIO.

7.1 SWOT analysis

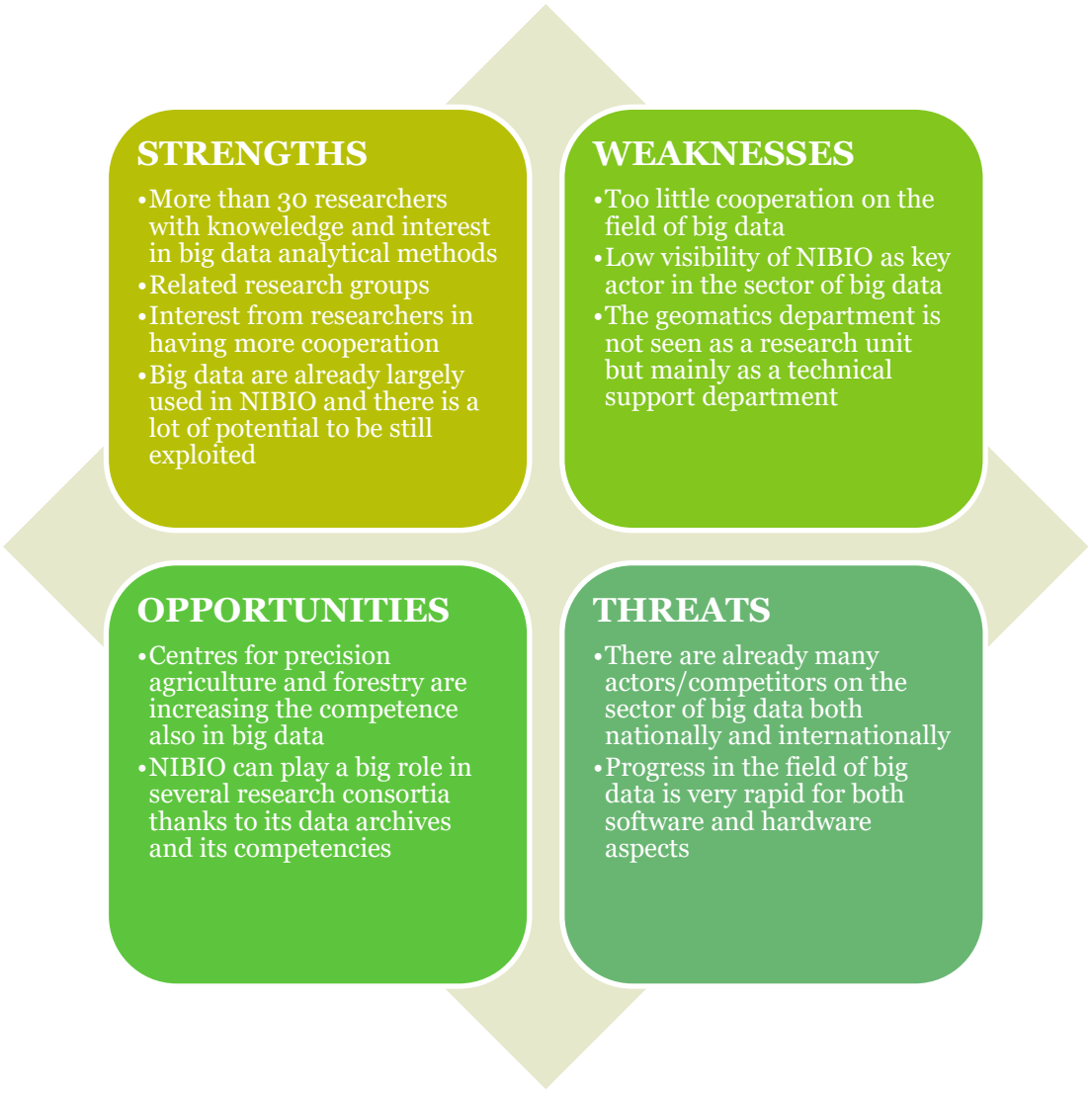

Figure 10. SWOT analysis for spatial big data at NIBIO.

7.2 Risks, criticalities, and bottlenecks in spatial big data

The use of spatial big data brings benefits, but also some risks and criticalities that should be considered carefully. Further, there are some elements that can represent bottlenecks in the use of spatial big data. In this chapter there will be a short discussion of these elements.

Strengths

Description	What to do
Availability: NIBIO has large volumes of well-managed data and access to more through the NSDI More than 30 researchers with knowledge and interest in big data analytical methods	Continue the focus on data management and documentation. Prepare data foreseen analysis Create network and arenas for cooperation and exchange of information
Interest from researchers in having more exchanges	There should be occasion where researchers working with big data can meet to discuss ongoing issues and future opportunities and challenges.
Big data are already used in NIBIO and there is a lot of potential to be still exploited	Have more occasion of exchanges experiences and knowledge (both virtually and physically). Develop a strategy for Big Data research in NIBIO

Weaknesses

Description	What to do
Too little cooperation between units	Create more occasion of discussions and identification of common needs and available skills, both virtually and physically.
Low visibility of NIBIO as key actor in the sector of big data	Increase communication activities on the sector of big data, both as scientific publication and more popular reports/articles.
The geomatics department is not a research unit and is seen as a technical support department by some researchers	Try to have more contacts with other divisions who might benefit from cooperation

Opportunities

Description	What to do
Centres for precision agriculture and forestry are increasing the competence also in big data	Continue the good work
NIBIO is an attractive partner for research consortia thanks to its data archives and its competencies	Improve internal communication initiatives to inform researchers about the opportunities represented by available datasets and internal competences in the sector of big data.

Threats

Description	What to do
There are already many actors/competitors on the sector of big data both nationally and internationally	Keep an overview of what other actors are doing and avoid going in directions where there is no more space for other actors, avoiding also to "reinvent the wheel". Develop a Big Data research strategy for NIBIO and define the delimitation against other actors
Progress in the field of big data is very rapid for both software and hardware aspects	It is important to identify what NIBIO can do with internal resources and what need to be done with external resources (e.g., with consultants). It is fundamental, however, that in all relevant division in NIBIO there are people to understand big data methods and potential for its own division.

7.3 Important issues related to spatial big data at NIBIO

Privacy

In an era when data have a central importance and all information is becoming digital, privacy is becoming a central aspect to consider when dealing with personal or sensitive data. This is not only a concern for data related to people, but also for data related to companies. NIBIO handle many datasets related to the bioeconomy sectors, including socio-economic data from forestry, agriculture and farming, data coming from machines while working on the field, biometric information of operators and more. This information must be handled carefully since not all can be made available directly to the public. Depending on the purpose for what these data have been collected for, they might comply with different national or EU legislation. It is therefore important to have standard, shared and robust routines within NIBIO to prepare data management plans and implement them.

Handling of privacy issue cannot be done at one central point only within NIBIO, but different departments can have different roles that should be clearly identified and known by everybody dealing with data having potential privacy issues.

- **Document management/Geomatics:** prepare data management plans, useful for proposal and all project where data play a large role.
- **IT:** provide technical support for the implementation the plan.
- **All divisions:** provide information useful for the preparation of the data management plan and follow its pescriptions and suggestions during ordinary work.

Data ownership

One of the Vs of Big Data is Value, i.e., the value that big data has. Being owner of data has implications related to intellectual properties and licenses but also related to the economic potential benefits.

When releasing data out of the organization it is always important to do that clearly stating intellectual properties rights. Having a license covering the data clarifies in legally binding terms the possibilities (and limitations) on the use, further reuse, distribution, and modification of the data. The licence can, but does not necessarily associate, a cost with the data or make them unavailable for certain use or users. Researchers should be aware of the importance of protecting the intellectual property.

NIBIO has implemented an online archive system called BIRD to publish and distribute research datasets. The system can be used to assign a DOI to the datasets. However not only published data is

distributed and not all data delivered outside need to be published on the BIRD archive. In both cases it is important to associate a licence.

Using online platforms and cloud-based infrastructures

The use or processing of data through online platforms and cloud-based infrastructures gives many opportunities, including the possibility of speeding up large processing, elaborate large datasets and access to powerful facilities available only online. There are, however, implications related to data security, data ownership and, indirectly, also to data value. All online platform and cloud infrastructures adopt specific policies and licenses of use and all issues related to these aspects should be clarified. Adopted policies and licenses of uses can vary considerably based on the country where the owner of the solution is based. Norway, although not part of the EU, comply with many of the EU regulations related to data ownership and intellectual property. Such rules are generally quite safe and protect the owner of the data. Other solutions, such as Google, Amazon, or Microsoft, are based outside the EU, and do not comply with EU regulations. This can pose risks to data ownership.

The use of Google Earth Engine, as an example, can give multiple benefits: e.g. easy access to large datasets and powerful and rapid processing of satellite data. But this comes with a cost. The code run on Google Earth Engine will be available for Google, and if a code is open for everyone, should not be uploaded on that platform. However, this does not mean that its use should be always discouraged. Simple operations on large datasets which are easily available on Google Cloud, such as calculating simple statistic over time series of satellite images can be safely done on Google Earth Engine and results can be downloaded locally. This can reduce processing time and avoid the download of large dataset. Once downloaded, these pre-processed data can be used locally for more advanced processing without further issues regarding both data and code.

It is important that NIBIO researchers are aware of the implication of various solutions. It is a daunting task to have a continuously updated overview, but it would be advisable to place this responsibility in a single department. Further, this could help in having an overview of the used solutions to organise trainings, if needed, put researchers using the same solution in contact and provide better problem solving and support when needed.

Security

The previously described issue of data ownership is relevant for all data, but particularly for strictly reserved data such as those of the National Forest Inventory, AR5 and the 3Q program.

Protection of data should cover both storage, transfer and use of data. This is mainly the responsibility of the IT department, but researchers have a large degree of autonomy concerning the transfer and use of data. It is therefore important that security issues related to the chosen method for storing or transfer of datasets are considered by all parts of the organization, including researchers.

The geomatics department is already in charge of several services for distributing data and for tracing data transfer: these services can be important also for big data and additional services could be implemented based on future needs.

An important issue is posed in addition by interconnectedness of devices. The use of smart devices that are connected to the network and can transmit data is becoming more common also within NIBIO, and all these devices must be connected in a safe way, both to protect the transferred data and to protect the entire IT infrastructure that can become more expose to attacks.

Finally, some big data might need to be accessible and updatable from external entities: giving control of data to someone else poses problem of data integrity as well as safety, having the risk of making the entire system and IT infrastructure more vulnerable.

Use of blockchain in big geospatial data

The use of blockchain[18] is increasing rapidly and expanding across sectors. Blockchain can play an important role with respect to the use of spatial big data in the bioeconomy sector. In fact, being an immutable registry for transactions of digital tokens, blockchain is suitable for geospatial applications involving data that is sensitive or a public good, autonomous devices and smart contracts. Within NIBIO it can be useful for data related to agriculture and farming, including subsidies, forestry, and many others.

The introduction of blockchain at NIBIO requires a large effort in building competences, but can be beneficial for e.g., the centres for precision agriculture and precision forestry as well as people working with socio-economy aspects in agriculture and farming. A joint work involving people from IT, geomatics and other departments could allow this.

Ethics

Most of the data handled at NIBIO has little or no ethical implication, but some data, e.g., those related to monitoring of workers in the forestry sectors, can have ethical implication related to their use. Besides, data involving animals can also involve ethical issues. Furthermore, it is important that elaboration of data do not lead to wrong representation of the reality. It is therefore always important to have an ethical approach to the use of data (as it should be in any research carried on at NIBIO).

Bandwidth

Dealing with big data requires availability of IT infrastructure capable of performing rapidly the required analysis. While a lot of attention is on the server and other computing facilities used to process data and on the storage solution, there is not always enough attention on the physical infrastructure for transferring data. If data is not processed in the same machine where the data is stored, data might need to be transferred, at least for the time of execution of the analysis (the only exception can be when doing analysis on distributed storage sending the code to the place where data is stored and sending back only the results). Having slow physical cable connecting some of the used machine can result in bottlenecks that slow down the speed of large processes.

Having many stations, and with an increasing amount of work via home office, NIBIO needs an efficient way to process big data, and this is already done by having centralized servers with the possibility of having a remote access via SSH. It is therefore important to continue in this direction and push all researchers performing large analysis in using solution that can enhance the productivity, compared to analysis performed locally.

Dimensionality/Spatiality

The high dimension of data, that can have even very high resolution over large areas, is another important issue to consider and that need to be properly handled.

[18] A blockchain is a growing list of records, called blocks, that are linked using cryptography. Each block contains a cryptographic hash of the previous block,[6] a timestamp, and transaction data (generally represented as a Merkle tree). By design, a blockchain is resistant to modification of the data. It is an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way. For use as a distributed ledger, a blockchain is typically managed by a peer-to-peer network collectively adhering to a protocol for inter-node communication and validating new blocks. Once recorded, the data in any given block cannot be altered retroactively without alteration of all subsequent blocks, which requires consensus of the network majority. Although blockchain records are not unalterable, blockchains may be considered secure by design and exemplify a distributed computing system with high Byzantine fault tolerance. Decentralized consensus has therefore been claimed with a blockchain

7.4 Possible pathways of future developments in NIBIO

Creation of competence centre for spatial big data

It is neither useful nor efficient to centralize all activities related to big data. Still, a competence centre with this focus can have multiple benefits, such as:

- Improve data management across NIBIO, especially for data of interest for multiple departments;
- Provide technical support in relation to spatial big data;
- Organize trainings based on real needs;
- Organize meeting places and facilitate exchange of information and coordinate training activities;
- Increase the visilibity of NIBIO as centre having strong competences on big data;
- Advice on issues concerning property rights and data security
- Optimize activities related to spatial big data across divisions.

A competence centre would not be a centre in charge of performing everything related to spatial big data, however, can help performing more advanced analysis and can support in the preparation of proposals. Further it can support in the development of tools (and code) that can be beneficial for different divisions/methods, trying to coordinate the different activities and transfer knowledge acquired from experiences in one department to others (e.g., similar algorithm can be used for land cover mapping, agricultural mapping or forest mapping without starting every time from scratch, as is happening now where everyone does its own work independently.

The competence centre could also provide support to departments hiring staff with competence in that sector and help evaluating candidates who will work with big data methods.

The creation of a competence centre could be linked to ongoing project trying to avoid having additional costs for NIBIO.

Improvement of spatial big data infrastructures

The competence centre could be in charge of supporting the development of the internal infrastructure for spatial big data within NIBIO, by collecting and evaluating needs coming from the different divisions. The centre would also be in charge of facilitating the use of such infrastructure.

Furthermore, the centre could facilitate the cooperation with external infrastructure, and with Uninett Sigma2 in particular. Since nowadays the facility is more focused on non-spatial big data, it would be important to promote the relevant of providing better tools also for researchers working with spatial big data. In addition, there could be cooperation also with other institutional infrastructures such as those promoted e.g., by the EU or by ESA or at the Nordic level.

References

Cattell, R. (2011). Scalable SQL and NoSQL data stores. *ACM SIGMOD Record*, *39*(4), 12-27. https://doi.org/10.1145/1978915.1978919

Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. *Mobile networks and applications*, *19*(2), 171-209.

Commission, E. (2012). Innovating for sustainable growth: a bioeconomy for Europe. *Communication from the Commission to the European Parliament, the Council, the European economic and Social Committee and the Committee of the regions, Brussels*, *13*, 2013.

Dean, J., & Ghemawat, S. (2004). *MapReduce: simplified data processing on large clusters* Proceedings of the 6th conference on Symposium on Opearting Systems Design & Implementation - Volume 6, San Francisco, CA.

Dean, J., & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the Acm*, *51*(1), 107-113. https://dl.acm.org/citation.cfm?doid=1327452.1327492

Ghemawat, S., Gobioff, H., & Leung, S.-T. (2003). The Google file system. *SIGOPS Oper. Syst. Rev.*, *37*(5), 29-43. https://doi.org/10.1145/1165389.945450

Hilbert, M., & Lopez, P. (2011). The world's technological capacity to store, communicate, and compute information. *Science*, *332*(6025), 60-65. https://doi.org/10.1126/science.1200970

Hoyer, S., & Hamman, J. (2017). xarray: ND labeled arrays and datasets in Python. *Journal of Open Research Software*, *5*(1).

Iorga, M., Feldman, L., Barton, R., Martin, M. J., Goren, N. S., & Mahmoudi, C. (2018). *Fog computing conceptual model*.

Li, J., Liao, W.-k., Choudhary, A., Ross, R., Thakur, R., Gropp, W., Latham, R., Siegel, A., Gallagher, B., & Zingale, M. (2003). Parallel netCDF: A high-performance scientific I/O interface. Supercomputing, 2003 Acm/Ieee Conference,

Li, Y., & Manoharan, S. (2013). A performance comparison of SQL and NoSQL databases. Communications, computers and signal processing (PACRIM), 2013 IEEE pacific rim conference on,

Mahecha, M. D., Gans, F., Brandt, G., Christiansen, R., Cornell, S. E., Fomferra, N., Kraemer, G., Peters, J., Bodesheim, P., Camps-Valls, G., Donges, J. F., Dorigo, W., Estupinan-Suarez, L. M., Gutierrez-Velez, V. H., Gutwin, M., Jung, M., Londoño, M. C., Miralles, D. G., Papastefanou, P., & Reichstein, M. (2020). Earth system data cubes unravel global multivariate dynamics. *Earth Syst. Dynam.*, *11*(1), 201-234. https://doi.org/10.5194/esd-11-201-2020

Rew, R., & Davis, G. (1990). NetCDF: an interface for scientific data access. *IEEE computer graphics and applications*, *10*(4), 76-82.

Ryan, A., kevin, p., joe, h., matthew, r., chiara, l., michael, t., Naomi, H., Richard, S., Ryan, M., & Davide, D. V. (2017). *Pangeo NSF Earthcube Proposal*. https://doi.org/10.6084/m9.figshare.5361094.v1

Shvachko, K., Kuang, H., Radia, S., & Chansler, R. (2010). The Hadoop Distributed File System. Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST),

Uehara, M. (2017). Mist computing: Linking cloudlet to fogs. International Conference on Computational Science/Intelligence & Applied Informatics,

NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

NIBIO - Norwegian Institute of Bioeconomy Research was established July 1 2015 as a merger between the Norwegian Institute for Agricultural and Environmental Research, the Norwegian Agricultural Economics Research Institute and Norwegian Forest and Landscape Institute.

The basis of bioeconomics is the utilisation and management of fresh photosynthesis, rather than a fossil economy based on preserved photosynthesis (oil). NIBIO is to become the leading national centre for development of knowledge in bioeconomics. The goal of the Institute is to contribute to food security, sustainable resource management, innovation and value creation through research and knowledge production within food, forestry and other biobased industries. The Institute will deliver research, managerial support and knowledge for use in national preparedness, as well as for businesses and the society at large.
NIBIO is owned by the Ministry of Agriculture and Food as an administrative agency with special authorization and its own board. The main office is located at Ås. The Institute has several regional divisions and a branch office in Oslo.

Front cover: image by Gerd Altmann from Pixabay.
Back cover: image by Tumisu from Pixabay.

nibio.no