ANNEX 2
CONCEPT OF AGRICULTURAL BIG DATA SYSTEM

Agricultural big data project

tieto

Contents

1. Definitions and abbreviations

Definition	Explanation
ADS	Estonian address data system
Database	Dataset officially registered in RIHA or officially unregistered dataset
MeM	Ministry of Rural Affairs
PMA	Board of Agriculture
PMAIS	Information system of the Board of Agriculture
PMK	Agricultural Research Centre
PRIA	Agricultural Register and Information Board
RIA	Information Systems Authority
RIHA	Administration system for the state information system. This is a metadata system, which has to define information systems included in state information system.

Table 1. Definitions and abbreviations

2. Overview

In Estonia big data systems are being created in several areas. However, there are currently no established practices and standard concerning how these systems should be created. Moreover, there is no common agreement on what the big data are. Therefore, this project aims at cooperation between different administrative areas to find as comprehensive solution as possible, attempting to utilise all knowledge generated in the country so far.

For the purposes of this project, big data are data that comply with the following criteria:

1) data are added quickly;
2) data are very complex;
3) data have great volume;
4) data content is not known in detail or data are not structured.

The analysis did not find such data that would comply with all those criteria. Assessment criteria are certainly a bit soft and subjective, but when compared to the rest of the world, the volume of agricultural data generated in Estonia is rather low.

However, we managed to find future sources of big data, consisting mostly in data from agricultural machines or field work data, detailed data concerning ground and soil, and indexes calculated based on satellite images, when presented for each square metre of ground. Such data are not yet used on national level and there are also no systems to register and utilise them. First step seems to be systematic registration of field work data in electronic format and calculation of indexes based on satellite image, which characterise crop production.

The concept of his big data system is based on data analysis, legal analysis and economic analysis of agriculture-related databases, including analysis of potential big data system services.

Data in big data system are divided into the following categories:

1) Real data concerning objects and subjects. Big data system contains some real data that are not included in interfaced databases and that cannot be included in any other database. Additionally, it is possible to copy data from various databases to big data system when performing major analyses, in order to allow performing the analysis in real time and allow repeating it with supplemented algorithms.
2) Description of real data and services or metadata.

3) Information concerning individuals (system users) and their rights and authorisations.

Big data system will not permanently contain data of other databases, including spatial information, for which the Land Board has developed strong infrastructure and publicly used components, and there is no point in competing with that. This also applies to the environment for processing ESTHub satellite images, created by Land Board, capacity of which will be used for developing and calculating various indexes in future.

3. Parameters required for big data system

Big data system is divided into the flowing sections (environments):

Environment	Explanation
Data acquisition and processing environment	Environment for authenticated users intended for entering data in big data system. The task of data acquisition environment is to ensure that the data reaches big data system and are processed into format required for providing the services. Data processing may take place as background process or performed by user. In order to use background processes, a separate server with relevant software must be planned.
Analytics environment	Environment for authenticated users intended for analysing and using data for research purposes in order to identify real content and quality of data and prepare new data services. The services may include both information services regarding data in state registers and services that issue derived data, e.g. NPK and humus balance calculators, fertilising recommendations and fertilising maps. Output of analytics environment consists in visualisations of basic data and data quality and files (csv, xlsx), that shape data into a format suitable for the end-user. Visualisations represent electronic consumer services. Visualisations of spatial data and map files can be used as input for agricultural machines for performing work.
Web environment for services	Environment is divided into two: 1) environment with authenticated user; 2) open data environment with unauthenticated user. The environment with authenticated user operates as an environment for making person- and field-specific queries, where producers receive information about themselves and objects of their undertaking by using various big data system services. Big data system administrator will grant user rights if there are legal grounds that allow using data by that particular person. In environment with unauthenticated user, the users can view sector-specific open data and use services created based on open data.
Environment for big data system administration	Environment with authenticated user for operations performed by big data system administrators. Main activities include: 1) Administration of metadata or data definitions. Data definitions contain description metrics and dimensions and data models for databases and services. 2) Administration of classifications and code lists.

Environment	Explanation
	3) Technical administration of system database and applications. 4) Setting up, testing and monitoring of data entry forms. 5) Setting up and monitoring of big data system services. 6) Administration of user and right groups. 7) Real data acquisition from external sources, files and manual entry.

Table 2. Subsystems of big data system

Table describes main subsystems of the system.

1.1 Data acquisition and processing system

Data acquisition is moving data to big data system by manual or automated means. Big data system must allow not only data use but also data management, including data entry. Such data include e.g. information about pests, test results (EULS) or agrometeorology data, which are currently stored on paper, in Excel or text format.

There are three ways to enter data in the database of big data system:

1) acquisition of data from electronic source and its entry into database and;
2) entering data manually;
3) entering data files in database.

Big data system must support the said options. More detailed requirements for providing solutions for these operations are described below.

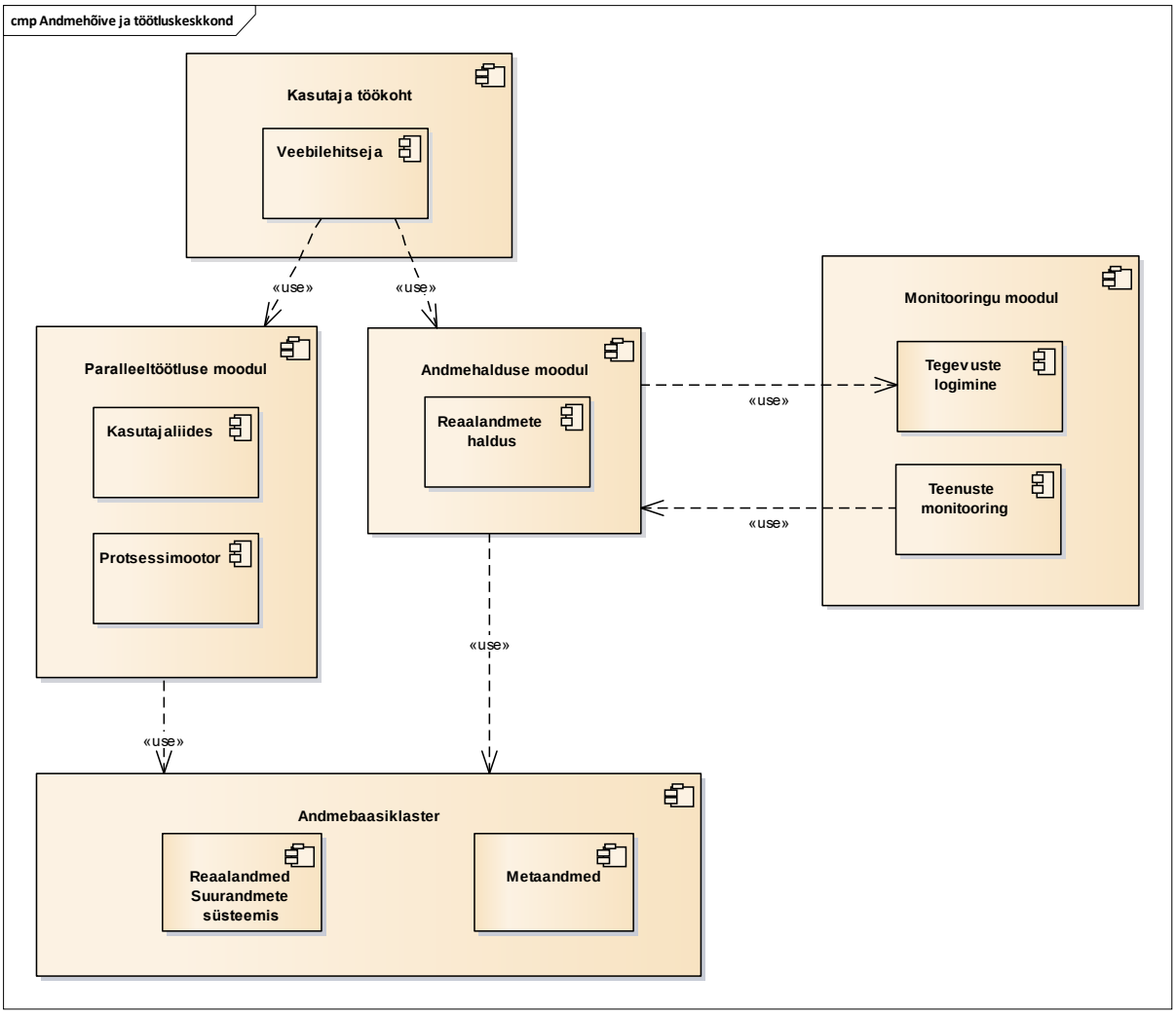

Figure 1. Data acquisition and components of processing environment

User who is not system administrator but can enter data in the system, can real data entry module in the system, including metadata. Moreover, the user has an opportunity to further process real data and make calculations to supplement dataset with additional parameters.

All operations in data acquisition and processing environments are logged and applications visible for users are monitored.

Main property of the data acquisition environment is that it can be adjusted for acquisition of data different from the management module. System allows creating user interfaces for such data that currently lack database or management software.

The following section explains the meaning of standard format of data and how it is possible to use data with different content within the same structure.

1.1.1 Standard format of data

One major problem when handling data consists in their different structure, undefined content and quality issues.

The data stored in big data system are converted to standard format. This concept is based on an assumption that data are generated in the course of an event – either soil analysis, measurement taken at the weather station, or measuring the quantity of harvested crop in a harvester. Various measurements

are taken, manually or automatically by using sensors. One event may generate 1-n parameters that may be either numerical or contain text. Content of the parameter is defined by metric. Event is always related to dimensions. Dimensions may also be specified later on, after event data have already been added to the database. Events of different types of datasets may vary. Event types must be defined as classification. Its elements (event types) must be linked to certain metrics. After adding real data we get a data structure that contains both real data and their metadata, all of which can be searched from the system and used in visualisations for user interface services and web services.

There are certain risks that should be considered in case of abstract structure of data. Depending on database engine, search may be slow in such system. Problem can be solved by means of transforming data or by using parallel processing. When it is necessary to perform large-scale analyses, data structure can be converted to other format, especially regarding the data to be analysed – to create analysis database, which allows more convenient and efficient use during analysis. That way, data become products that can be used both for services and in analytics environments.

Transformation always means delay in renewing data. Services and analyses must take that into consideration.

As for data processing, the following operations can be distinguished:

1) **Data integration and transformation** – combining data from several sources, converting data to a format suitable for use.
2) **Data classification and encoding** – textual data is attributed a code which classifies data entry (event).
3) **Data validation** – carrying out quality control and visualisation of its results.
4) **Data editing and imputation** – in case of data that lack certain values, which makes it impossible to use them, so-called imputed data are created by deriving them from other data or based on any other accepted algorithm. For example, in order to get a time series of a certain soil parameter for calculation purposes, one has to take the data of the most recent soil analysis and calculate the parameter for the subsequent period, taking into account the crop grown on the field, quantity and content of fertiliser used and other parameters required for detecting the missing value. If there are no other data, parameter must be taken from previous period and generate virtual imputed soil analysis event. Information of the event generated that way must be distinguished from other parameters by means of corresponding specifier.
5) **Derivation of new parameters** – calculating and defining additional parameters based on existing event.
6) **Data aggregation** – summing data to generalised level.
7) **Calculation of weights** – calculating weights for parameters or sums in case of sample data.
8) **Output formation** – formalising the processing result as a database or a data file.

Big data system must allow performing the above operations. It must allow setting up (programming) processing algorithms according to need.

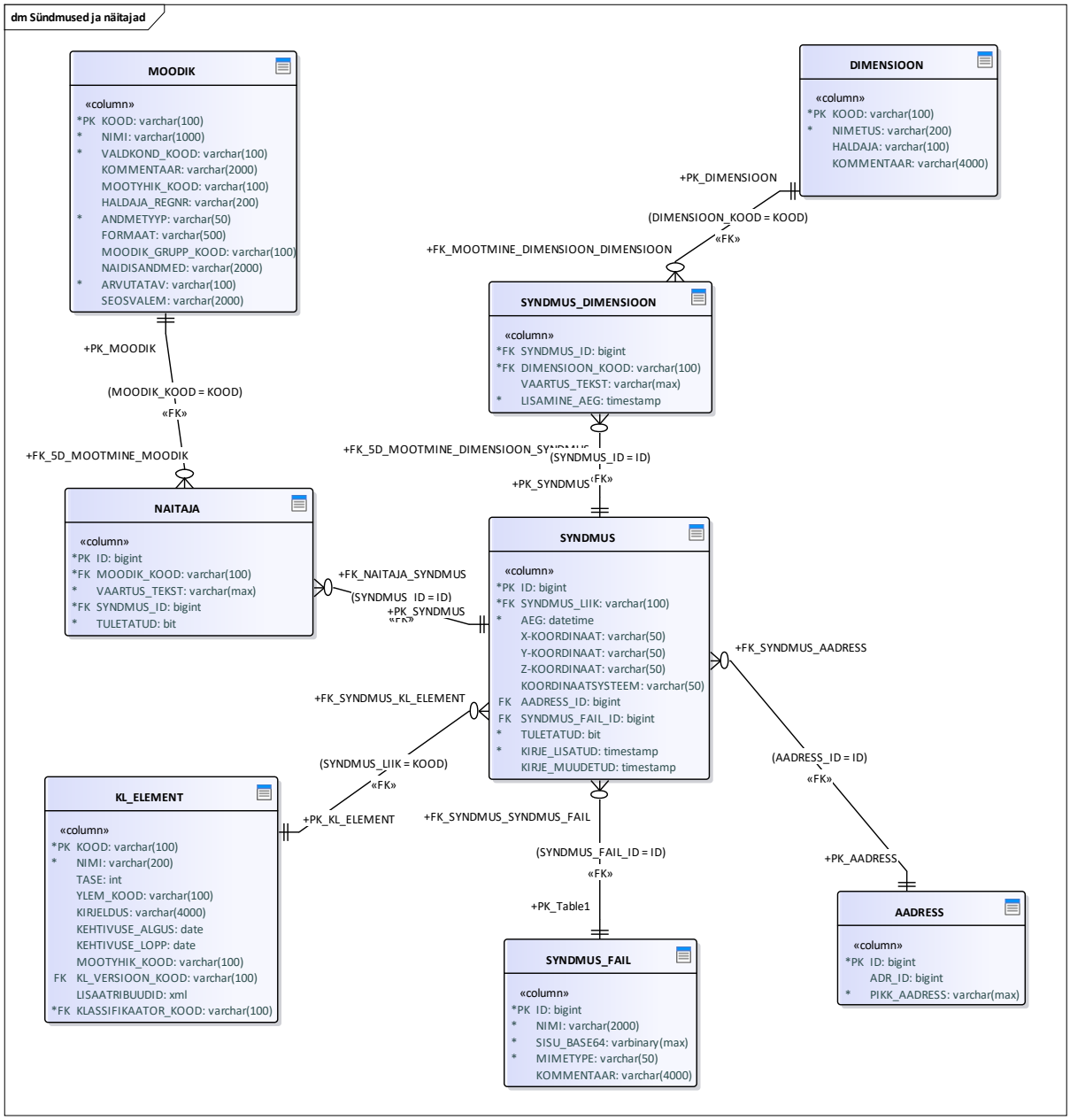

Figure 2. Standard data structure for different datasets

In order to manage the data shown in the figure, big data system must comply with the following requirements:

1) Data structures are defined by using event type classification, where every type corresponds to a certain number of metrics and dimensions that characterise the conditions for occurrence of events.
2) Definitions of event types must allow the system to generate user interface for entering data.
3) System must allow importing event and parameter data in XLSX and CSV file format. Upon importing a file, the system will verify data compliance with event type and definition of particular dataset.
4) System must allow storing all the data indicated in the figure above in the database. Keep in mind that entering of some data requires using GPS location of the device used for entering data and it must be possible to determine site coordinates by the map in the device.

5) In future, separate data entry forms will be created for entering data with more complex associations, but data must be stored in a standard data structure.
6) Big data system must allow materialisation of data so that the data are de-normalised to tables in their first normal form, where data of one dataset (parameters of all metrics and dimensions of an event type) are found in one table, each cell contains one value and output is issued to the user as a single table in xlsx or csv format. Upon issuing data, the user must be able to filter the events by type, period, metrics and dimensions, so that data are issued in the number of columns requested by the user per event type.
7) System must allow retrieving data from the system based on a single metric, so that the output consists in all real data across all data sources.
8) Big data system must allow performing data quality control.
9) Data must allow processing, in the course of which extracts are generated from data. Basic data will remain in the database in the same format as they were received.

1.1.1.1 Parameters

Table NAITAJA shows the outcome of measurement, i.e. parameter that can be either measured temperature, measured volume or weight of crops, measured quantity of fertiliser or verification of compliance with a requirement.

Primary key	Attribute	Data type	Mandatory	Description
True	ID	bigint	True	Unique measurement identifier.
False	MOODIK_KOOD	varchar(100)	True	Metric reference. More detailed content of the parameter is described at the metric. Metric enables access to entire metadata from essential explanation of the metric to classifications.
False	VAARTUS_TEKST	varchar(max)	True	Value can be either a certain measured value or reference to classification element, if measured result can be expressed on a scale defined as classification (code list). For example, upon quality measurement, the values may include "low", "medium" or "high". In case of text, this may contain a longer unstructured value, e.g. error message received during measurement.
False	SYNDMUS_ID	bigint	True	Reference to event.
False	TULETATUD	bit	True	Shows whether parameter data are derived based on other data. Derived when absolutely necessary, if missing information does not allow making calculations, queries or providing a service.

Table 3. Data structure of parameters

1.1.1.2 Events

Table SYNDMUS registers an event, during which certain parameters were identified. In case the event or parameter must be equipped with textual explanation, one has to use parameter and metric expressing relevant explanation.

Primary key	Attribute	Data type	Mandatory	Description
True	ID	bigint	True	Unique event identifier.
False	SYNDMUS_LIIK	varchar(100)	True	Reference to event type. Event type must be defined as classification. Set of possible parameters of the event must also be defined as classification, where corresponding event dataset link must be generated between event type and data element.
False	AEG	datetime	True	Time of event.
False	X-KOORDINAAT	varchar(50)	False	Measurement site X-coordinate.
False	Y-KOORDINAAT	varchar(50)	False	Measurement site Y-coordinate.
False	Z-KOORDINAAT	varchar(50)	False	Measurement site altitude.
False	KOORDINAATSYSTEEM	varchar(50)	False	Coordinate system indication according to ISO, where coordinates are expressed in EPSG:nnnn format. See https://spatialreference.org/ref/epsg/ .
False	AADRESS_ID	bigint	False	Reference to measurement site address.
False	SYNDMUS_FAIL_ID	bigint	False	File ID, where the event data came from.
False	TULETATUD	bit	True	Shows whether event data are derived based on other data. Derived when absolutely necessary, if missing information does not allow making calculations, queries or providing a service.
False	KIRJE_LISATUD	timestamp	True	Time of adding entry.
False	KIRJE_MUUDETUD	timestamp	False	Time of changing entry.

Table 4. Data structure of events

1.1.1.3 Link between event and dimension

Table SYNDMUS_DIMENSIOON stores the links between event and dimension. For example, what substances or what devices were used with regard to the event.

Primary key	Attribute	Data type	Mandatory	Description
False	SYNDMUS_ID	bigint	True	Measurement event reference.
False	DIMENSIOON_KOOD	varchar(100)	True	Reference to dimension.
False	VAARTUS_TEKST	varchar(max)	False	Dimension value at the moment of the event. E.g. field identifier.
False	LISAMINE_AEG	timestamp	True	Time when dimension was added to event.

Table 5. Data structure of the link between event and dimension

1.1.1.4 Event data files

File(s) generated as a result of the measurement are recorded in table SYNDMUS_FAIL.

Primary key	Attribute	Data type	Mandatory	Description
True	ID	bigint	True	File's unique identifier.
False	NIMI	varchar(2000)	True	File name in original source.
False	SISU_BASE64	varbinary(max)	True	File content in base64 coding.
False	MIMETYPE	varchar(50)	True	File type.
False	KOMMENTAAR	varchar(4000)	False	Comment on file content.

Table 6. Data of data files linked to events

1.2 Running parallel processes on the background

Large-scale data analyses and processing may take a very long time. They require a separate server and cluster, where both data and processes can be divided into smaller sections and run as parallel processes. Major data that may need parallel processing are e.g. meteorological data, ground altitude model mapped by using Lidar by Land Board, also data retrieved from or sent to machine, when used in their original form. Also, indexes created based on satellite images if their resolution level corresponds to grid cell with length of 1-3m.

Parallel processing requires programming of processing in relevant environment. Different possibilities are discussed in the chapter on architecture variants.

1.3 Analytics environment

Analytics environment is intended for:

1) researchers for analysing data;
2) statisticians for analysing data;
3) preparing services and evaluating data in terms of their quality and content;
4) system developers for introducing data and testing possible solutions.

There is no point in creating software for analytics environment from scratch. Instead, it is reasonable to use existing readymade analytics software. The components of analytics environment are described in the figure below:

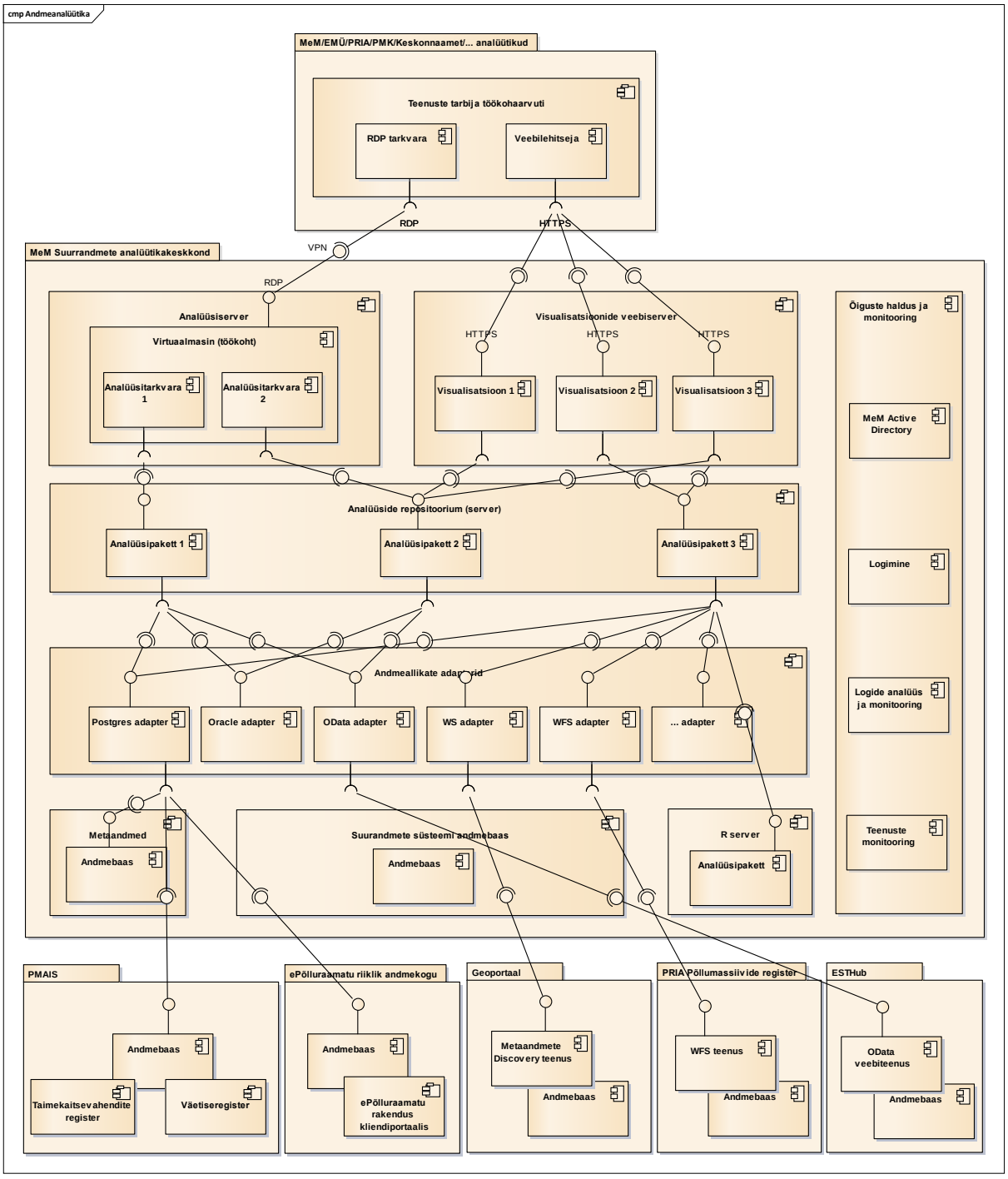

Figure 3. Components of analytics environment

The figure shows one possible configuration of analytics environment. Analytics environment must allow interfacing with data sources through adapters. Most widespread types of analysis software are equipped with adapters that support commonly used databases (Oracle, Postgres, etc.). They also support web services. As for web services, big data system has introduced Open Data standard (see [5]). As for spatial data, it uses WMS, WFS and WCS services. Web services represent preferred connection option.

Besides external connections, big data system uses internal data, i.e. metadata and real data regarding phenomena that currently lack separate database. Big data system will take the role of database for those data.

Analyst can use the analytics environment either through web browser or by entering the analytics environment and using *desktop* application. Specific feature of analysis software consists in extensive memory use when performing large-scale analyses, because calculations are made using data stored in memory. This means that PC memory use may easily exceed 8GB available in a PC with typical configuration. The machine used for performing the analysis should have at least 32GB memory. In order to ensure more efficient resource utilisation, it is reasonable to provide analyst with a virtual *desktop* machine with enough memory resource.

Different analyses and visualisations often require using several applications. For example, a software intended for reporting (BI) might not be suitable for analysing spatial data and vice versa – software intended for analysing spatial data (GIS) may encounter difficulties in reporting.

Analytics software of big data system deploys readymade software packages that come with a large number of complete adapters and visualisation tools. It must be possible to develop any missing functionalities regarding adapters, analysis packages and visualisation components. Developments must use open source code, including that of adapters. For more complex analyses, the software must allow using common R and Python scripts. Installed analysis software must not hinder using additional software. Big data configuration contains certain software, but analysts can use other software with equivalent adapters.

1.4 Services web environment

Service environment of big data system is used by agricultural producers and other persons with legitimate interest, who can access the data related to them by means of authenticated environment.

In addition to personalised data there are services that do not require authentication, e.g. services for sharing information in various public registers.

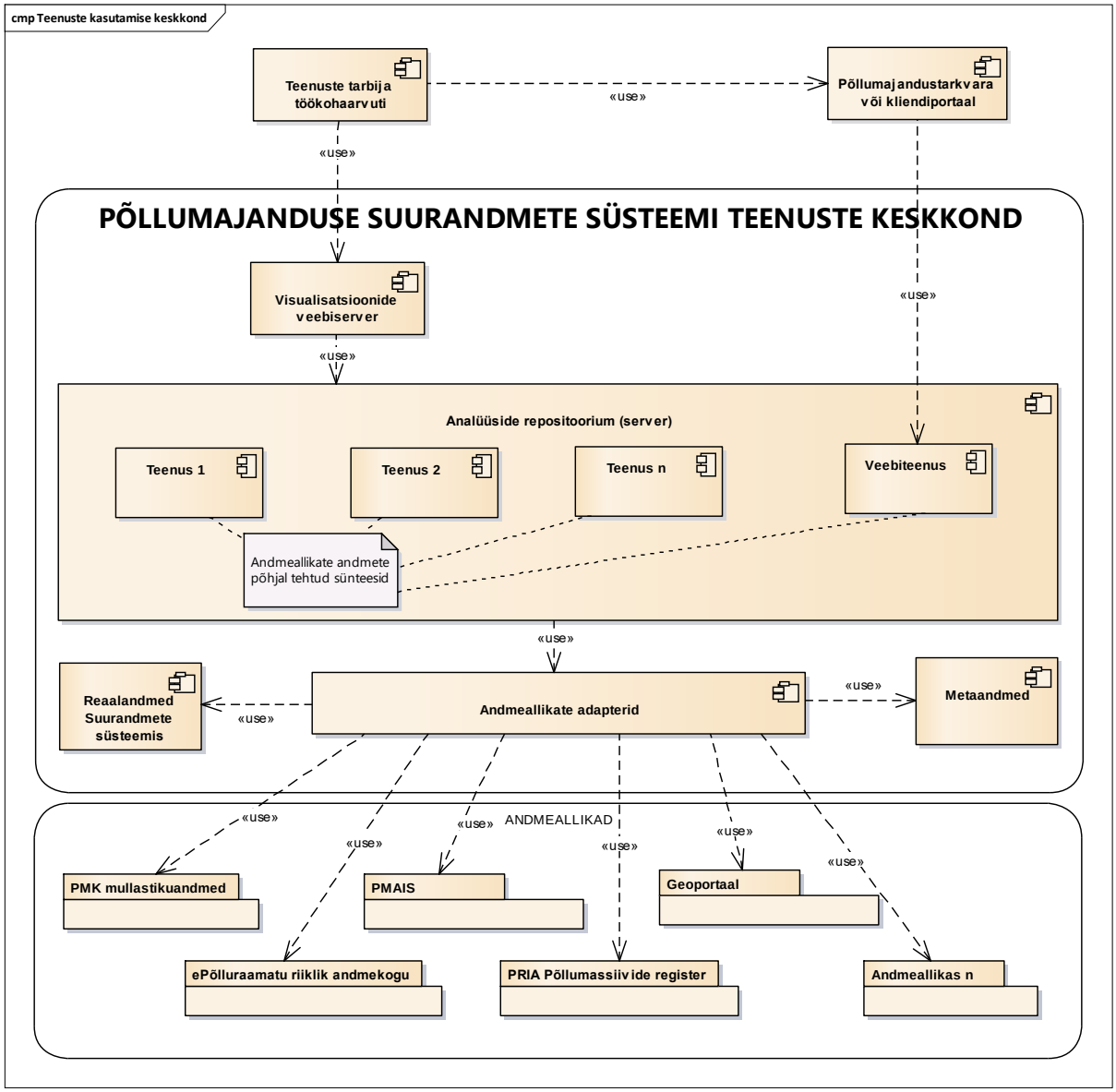

Figure 4. Service environment of big data system

The figure shows basic outline of service environment and it does not include all data sources or services. More detailed description of the services of big data system is provided in the chapter on services.

1.5 Management environment of big data system

Big data system administrator uses separate environments for managing big data operations.

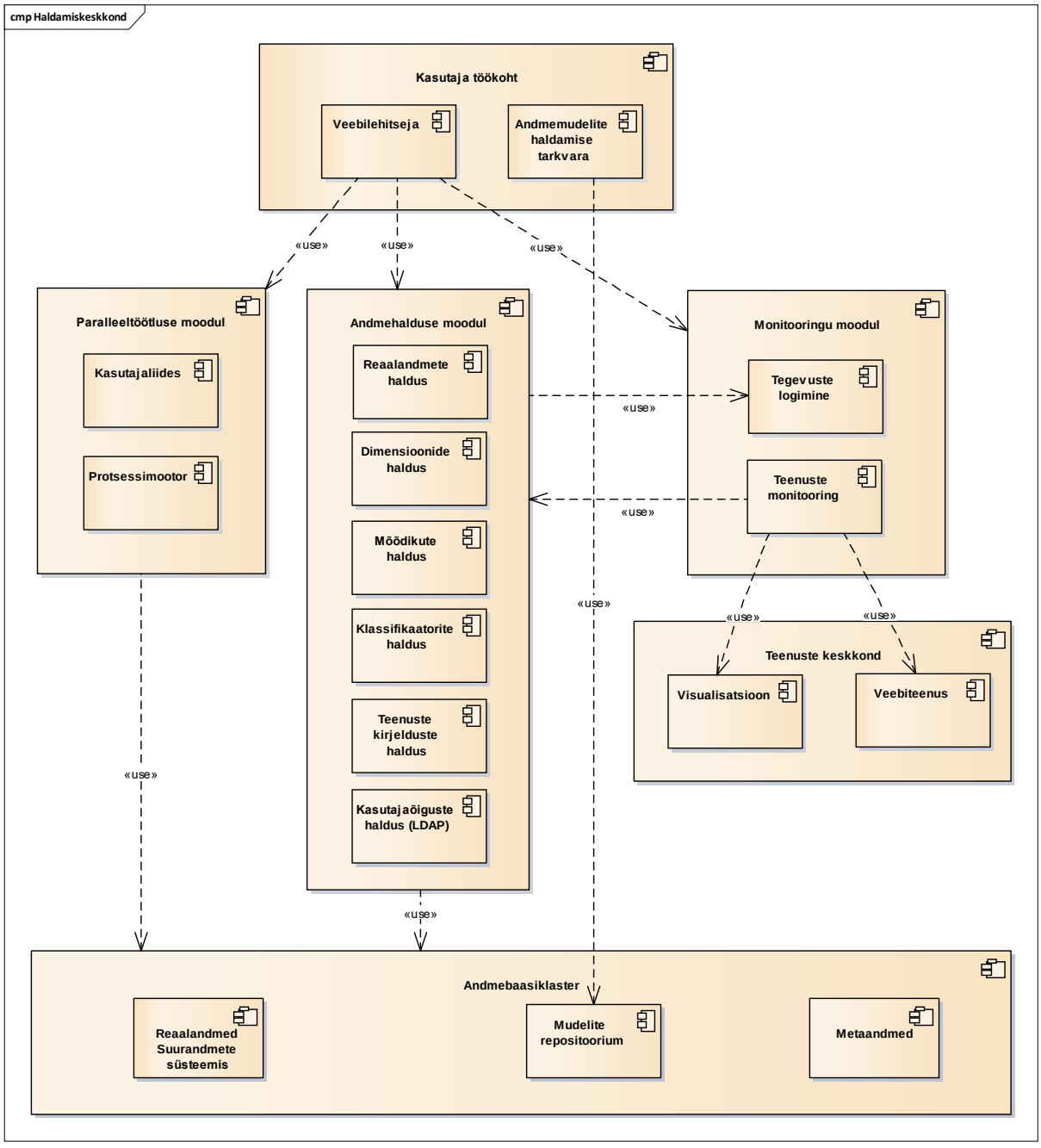

Figure 5. Management environment of big data system

Management solution is divided into the following sub-environments:

1) Parallel processing module – module that allows programming and running parallel processing and data analyses. This is a solution separate from data management.
2) Data management module – intended for managing system metadata and real data included in big data system.
3) Monitoring module – here you can set up monitoring of all services of big data system and trace data obtained from monitoring. It also stores logs. Logging requires a separate service that is used by other subsystems. Logs are stored as text files, so they do not depend on applications.

Detailed description of the requirements for management environment is presented in the chapter on requirements.

1.6 List of requirements for big data system

The table below contains functional and non-functional requirements for analytics environment.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-1	1. Metadata management	Classifications management	System must allow adding a classification by entering the name of the classification code and other attributes defined in data model. It must also allow other elements besides the code. Changing a code means adding a new classification.
REQ-2	1. Metadata management	Management of classification versions	Classification must allow creating a new version. Creation a version means creating new classification with the same code, while essentially adding a new classification and making a copy of elements of previous version. Classification version cannot be created from continuous time continuous classification.
REQ-3	1. Metadata management	Comparison of classifications	System must allow comparing two classifications. The result of the comparison consists in the comparison of classification field values and classification elements, bringing out similarities and differences. Comparison must be "smart" and able to relate classification elements with minor differences in spelling. Comparison must compare elements based on names and codes and in case of matching codes but different names, indicate the relation as similarity.
REQ-4	1. Metadata management	Management of classification elements	System must allow adding, changing and deleting of classification elements. the number of element attributes is determined in the data model.
REQ-5	1. Metadata management	Management of version of classification elements	System must allow managing the versions of classification elements so that in case of changing the element data, previous status is stored as separate entry and marked as invalid.
REQ-6	1. Metadata management	Management of relations between elements	System must allow creating hierarchies and networks of classification elements. Network means that one element may have 1...n parents. Hierarchy is formed if each element has 0...1 parent and 1...n children. It must allow relating elements of different classifications for compiling relationship tables, which serve as a basis for coding datasets from one classification to another.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-7	1. Metadata management	Management of element relation types	System must allow managing types of relations between elements. When creating a relation of classification element, the direction and type of the relation is also determined (parent child, part, etc.).
REQ-8	1. Metadata management	Sequencing of elements	System must allow determining classification element sequences within one classification. Elements may have several sequences within one classification.
REQ-9	1. Metadata management	Dynamic structure of elements, adding attributes	The number of attributes of classification element (data fields) must not be limited. System must allow not only adding mandatory element attributes, but also adding random number of data fields to classification element. Composition of data fields is valid within one classification.
REQ-10	1. Metadata management	Copying classification	System must allow copying the content of classification so as to create a new classification that has the content of the copied classification but new classification code.
REQ-11	1. Metadata management	Copying classification structure	It must be possible to use the description of the data composition of classification elements when creating a new classification. Here one must distinguish between copying of classification and copying of classification structure.
REQ-12	1. Metadata management	Description of data models, UML	System must allow managing data structure description as UML data model (ERD).
REQ-13	1. Metadata management	Import and export of data models	System must allow importing and exporting data models (and other UML models) to and from the system. Data exchange format is the most recent version of XMI (http://www.omg.org/spec/XMI/2.5.1).
REQ-14	1. Metadata management	Reverse engineering of data model from database, Oracle	Data model management system must allow reverse engineering of database model based on Oracle database. This means updating data model based on database.
REQ-15	1. Metadata management	Reverse engineering of data model from database, Potgres	Data model management system must allow reverse engineering of database model based on Postgres database. This means updating data model based on database.
REQ-16	1. Metadata management	Description of data models as UML data model	System must allow creating and managing data models in UML format.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-17	1. Metadata management	Generating data definition language (DDL)	System must allow generating data definition language based on data model so that the language can be run in database server.
REQ-18	1. Metadata management	Defining data	System must allow managing metadata of data sources (see data model).
REQ-19	1. Metadata management	Defining data source properties	System must allow defining metadata of databases in terms of services so that the definition would allow determining where and how the service can be used and what is its content.
REQ-20	1. Metadata management	Defining metrics	System must allow managing definitions of metrics. Metric is a data element, the value of which describes certain property of a particular object or phenomenon that can be described by using a metric.
REQ-21	1. Metadata management	Defining relation formulae between metrics	System must allow defining relations between metrics, including relation formulae of metrics.
REQ-22	1. Metadata management	Defining dimensions	System must allow defining dimensions. By dimensions, we mean non-measurable aspects describing the data object such as location, type, etc.
REQ-23	1. Metadata management	Defining relations between metrics and dimensions	System must allow managing relations between dimensions and metrics. Particular relation shows the conditions, under which particular parameter (metric value) of the dimension occurred.
REQ-24	2. Data integration	Defining data structure	System must contain data entry module with dynamic structure (can be changed by administrator), so as to allow generating data structures for entering different data (e.g. plant pests detection data, one part of which consist in the image taken in the field).
REQ-25	2. Data integration	Defining entry forms	System must allow a possibility of defining data entry forms, so that system administrator could create different forms for users to forward data to server.
REQ-26	2. Data integration	Defining controls	Data entry forms must allow defining controls applied when entering the data. It must allow controlling a certain value, range of values in case of numbers, absence of value, conditional absence of value, including by using all arithmetic operations.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-27	2. Data integration	Using REST service	System must allow data entry form to use REST services for making queries about data that the user can use for the entry form.
REQ-28	2. Data integration	Using data from X-Road service	System must allow linking data entry form with X-Road service so that after the user has entered the data, the system will prompt X-Road service. E.g. when user enters personal identification code, then system makes query about user's personal data from the Population Register and displays them on the screen.
REQ-29	2. Data integration	Using data from WFS web services	System must allow filling in data entry forms by using geoinformation from WFS service. For example information on fields from field register.
REQ-30	2. Data integration	Using data from WCS web services	System must allow filling in data entry forms by using geoinformation from WCS service.
REQ-31	2. Data integration	Using data from WMS web services	System must allow filling in data entry forms by using geoinformation from WMS service.
REQ-32	2. Data integration	Using data from database (Oracle)	System must allow using data from Oracle database.
REQ-33	2. Data integration	Using data from database (Postgres)	System must allow using data from Postgres database.
REQ-34	2. Data integration	CSV data acquisition (import)	System must allow using data from CSV file (data import).
REQ-35	2. Data integration	XLSX data acquisition (import)	System must allow using data from XLSX file to the extent of at least one worksheet without volume restriction.
REQ-36	2. Data integration	Storing data in database (Postgres)	System must allow storing data in Postgres database.
REQ-37	2. Data integration	Data queries and using GPS	Data entry forms defined by system must allow user to automatically use the GPS sensor in user's device so that the location given by sensor is stored with the data entered by the user.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-38	2. Data integration	Using photos on user's device	System must allow data entry form to link to the entered data with photos made by the user and accessible on user's device.
REQ-39	2. Data integration	Using files on user's device	System must allow data entry form to link to the entered data with data entered by the user and files accessible on user's device.
REQ-40	2. Data integration	Using direct queries	System must allow using data source data in real time without copying or buffering them. User must have an option to change data use settings either to buffered or real time.
REQ-41	2. Data integration	Data query from REST service	System must allow using data from REST services.
REQ-42	2. Data integration	Data query from X-Road service	System must allow using data from X-Road service.
REQ-43	2. Data integration	Data query from WFS web services	System must allow using data from WFS services.
REQ-44	2. Data integration	Data query from WCS web services	System must allow using data from WCS services.
REQ-45	2. Data integration	Data query from WMS web services	System must allow using data from WMS services.
REQ-46	2. Data integration	Data query from database, ODBC	System must allow using data from any database that has ODBC or JDBC connectivity.
REQ-47	2. Data integration	CSV data acquisition	System must allow using data from CSV file.
REQ-48	2. Data integration	XLSX data acquisition	System must allow using data from XLSX file.
REQ-49	2. Data integration	Data query from database, Postgres	System must allow using data directly from Postgres database.
REQ-50	2. Data integration	Data query from database, Oracle	System must allow using data directly from Oracle database.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-51	2. Data integration	Using data from database, Hadoop HDFS	System must allow using data directly from HDFS.
REQ-52	2. Data integration	File acquisition	System must allow transporting files and store them in system server.
REQ-53	2. Data integration	Reading messages from CAN BUS (ISO BUS)	System must allow reading data from CAN bus used in agricultural machines (ISO 11898).
REQ-54	2. Data integration	Applying data analysis to messages	System must allow processing information from CAN bus when read directly from PC COM port without intermediate software, except for operation system.
REQ-55	2. Data integration	Message filters	System must allow filtering CAN bus messages so that the user can retrieve only the messages he has requested.
REQ-56	2. Data integration	Creating message chains	System must allow creating chains of CAN bus messages, joining the chain of messages with pre-set signatures.
REQ-57	3. Data processing	Programming of processing	System must allow creating data acquisition and processing packages and use them to create workflows.
REQ-58	3. Data processing	Testing of processing	System must allow testing of created data processing packages and workflows.
REQ-59	3. Data processing	Timing of processing	System must allow running user-made packages and workflows according to pre-set timing.
REQ-60	3. Data processing	Periodical timing of processing	System must allow periodical timing of running packages and workflow.
REQ-61	3. Data processing	Version management of processing	System must preserve the history of changes in packages and workflows (versioning).
REQ-62	3. Data processing	Cancellation of processing	System must allow cancelling the processing made by the package if it is not followed by new processing.
REQ-63	4. Data analysis	Automatic profiling	System must allow automatic data profiling that would point out value frequencies per column, number of empty values, maximum and minimum values.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-64	4. Data analysis	Adhoc analyses (queries, generating queries)	Data analysis software must allow making freely chosen queries based on data, so that the query QSL is not generated by the analyst, but by analysis software. However, there must be a possibility to change the generated query manually.
REQ-65	4. Data analysis	Analysis and visualisation of spatial data	System must allow performing operations with spatial data, including placing data on map by address and geographical coordinates, find the area of geometry, find coordinates of the centre of geometry, find and display buffer zones with pre-set size, find overlap of geometries, find whether one shape is within the other and find geometries adjacent to the geometry in question.
REQ-66	4. Data analysis	Storing profiling results	System must allow storing the data profiling results so that it does not have to recalculate everything when displaying the profile.
REQ-67	4. Data analysis	Defining simple controls	System must allow defining data controls regarding the data in question in a manner that controls are not written in SQL but are similar to the language used for office software (e.g. Excel), which can be read and understood by an average PC user. At the very least, it must support all arithmetic operations, string operations for retrieving information from texts and exponentiation.
REQ-68	4. Data analysis	Defining complex controls across several databases and tables	System must support cross-usage of data and applying controls to the data in several tables and several data sources simultaneously.
REQ-69	4. Data analysis	Storing control results	System must allow storing data control results so that they are not re-calculated every time when viewing data.
REQ-70	4. Data analysis	Generation quality report	System must allow visualisation of the results of data control so that it displays statistical summary as well as all entries that did not pass the control. It must allow browsing entries based on controls.
REQ-71	4. Data analysis	Sharing of quality report online	System must allow sharing quality report as a webpage.
REQ-72	4. Data analysis	Compiling data model across different sources	System must allow compiling data model underlying the analysis by using several data sources, including several data sources of different types.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-73	4. Data analysis	Option to write transformation queries	System must allow performing data transformation in suitable format so that data tables can be merged and separated, columns merged and split.
REQ-74	4. Data analysis	Calculation of frequencies	System must allow calculating the frequencies of using values.
REQ-75	4. Data analysis	Clustering based on various features	System must contain clustering function.
REQ-76	4. Data analysis	Filters	System must allow interactive filtering of data both when creating analyses and visualisations and when using visualisations.
REQ-77	4. Data analysis	Forecast	System must contain data-based forecast function. System must allow the user to add forecast function.
REQ-78	4. Data analysis	Analysis programming	System must allow compiling data analysis packages by using R and Python.
REQ-79	4. Data analysis	R support (analysis programming)	System must allow using R for data analyses.
REQ-80	4. Data analysis	Option to use database and web service adapters	System must contain adapters for retrieving data from databases and web services. It must have adapters for at least Oracle, Postgres, MS SQL Server, Access databases, Open Data web services, XLSX, CSV, JSON and XML files and ODBC.
REQ-81	4. Data analysis	Performing in-memory analysis	System must allow running analyses in PC memory so that all analysed data are read in memory simultaneously.
REQ-82	4. Data analysis	Running analysis on application server	System must allow running analyses on application server. E.g. R server.
REQ-83	4. Data analysis	Running analysis on database	System must allow running analyses on database as SQL query.
REQ-84	4. Data analysis	Creating visualisations	System must allow creating data-based interactive visualisations.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-85	4. Data analysis	Creating visualisation templates	System must allow creating visualisation templates.
REQ-86	4. Data analysis	Visualisation of relational data	System must allow analysis and visualisation of relational data so that visualisations automatically take into account database relations and filter other related tables when filtering one table.
REQ-87	4. Data analysis	Visualisation of waypoint data on map	System must allow visualisation of data on the map by using geographic coordinates. It must support at least WGS84 coordinate system.
REQ-88	4. Data analysis	Extrapolarisation	System must support extrapolarisation of data. For example, in case a certain value is known regarding one or several points in a field, it must be possible to extend these values on the map to entire field, so that the area near the points is visually displayed in value-specific colour up to the border of the field.
REQ-89	4. Data analysis	Visualisation of analysis results on map	System must allow visualisation of map data by using address and geographic coordinates. It must also support using and displaying spatial form.
REQ-90	4. Data analysis	Simultaneous use of different map layers	System must allow using several map layers simultaneously.
REQ-91	4. Data analysis	Use of different base maps	System must allow using different base maps.
REQ-92	4. Data analysis	Online search of map layers	System must allow using map layers from WMS, WFS and WCS services.
REQ-93	4. Data analysis	Reporting	System must allow issuing data in report format.
REQ-94	4. Data analysis	Granting access rights to reports	System must contain user rights system, where access to reports and visuals takes place based on user and user group.
REQ-95	4. Data analysis	Printing reports (pdf)	System must allow issuing data in pdf format.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-96	5. Data sharing	Data sharing in csv format	System must allow issuing data in csv file.
REQ-97	5. Data sharing	Generation of data sharing web services	System must allow generating web services for data sharing without necessary system developments.
REQ-98	5. Data sharing	Sharing data in file format, file generation	System must allow using data to generate files and automatically store them in web server without necessary system developments.
REQ-99	5. Data sharing	Online sharing of visualisations and reports	System must allow online publication of data visualisations (services) compiled by system administrator, including for specific named users without public access.
REQ-100	5. Data repository	Storing and indexing relational data	System must allow storing data in relational form.
REQ-101	6. Data repository	Storing of unstructured data, including files	System must allow storing data files and reading them by analysis packages.
REQ-102	6. Data repository	Storing and searching XML data	System must allow storing data as xml documents and use such stored data in analyses.
REQ-103	6. Data repository	Statistic and diagnostics of data repository	System must contain data repository management module, which allows viewing collected data statistics and run diagnostics of load and performance during operation.
REQ-104	6. Data repository	Volume scaling of data repository with maximum interruptions of 15 minutes	System must allow extending database storage space without interrupting system operation for more than 15 minutes.
REQ-105	7. User authorisation	User registration	The system must allow user to register as system user. Registration is activated after its confirmation by system administrator.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-106	8. User authentication	User authentication by username and password	The system must allow user to log in by using username and password.
REQ-107	8. User authentication	User authentication by ID card	The system must allow user to log in by using ID card.
REQ-108	8. User authentication	User authentication by Mobile ID	The system must allow user to log in by using mobile ID.
REQ-109	8. User authentication	User authentication by SmartID	The system must allow user to log in by using SmartID.
REQ-110	8. User authentication	Password reminder	System must contain password reminder function.
REQ-111	8. User authentication	Two-factor authentication	System must contain two-factor authentication, which means that after login, the system sends to user a security code that has to be entered as second password. It must be possible to enable and disable two-factor authentication.
REQ-112	7. User authorisation	Granting user permission to enter the system	System must allow granting or removing user permission to enter the system.
REQ-113	7. User authorisation	Granting user right of access to an object	System must allow granting user rights to use the object in the system. Objects may include data tables, analysis packages, visualisation files and system functionalities. Enabling and disabling of functionalities means downloading and uploading data, downloading and uploading analysis packages and downloading and uploading visualisations.
REQ-114	9. System monitoring	Monitoring of web services	System must allow setting up monitoring of interfaced services. The function of monitoring is to detect whether the service is operational.
REQ-115	9. System monitoring	Monitoring of user interface	System must allow setting up monitoring of operation of user interface of different parts of the system. The function of monitoring is to detect whether user interface is operational.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-116	9. System monitoring	Logging user activities	System must log all system logins. The system must also log all changes in user rights.
REQ-117	10. System administration	User management	System must contain a solution for user rights management or allow interfacing with LDAP.
REQ-118	10. System administration	User groups management	System must contain user group management or interface with a LDAP that contains user group management.
REQ-119	10. System administration	Group rights management (authorisation)	System must include granting rights to user groups or be interfaced with LDAP that allows such management.
REQ-120	10. System administration	Servers management	The solution must all using several servers and system configuration must include administrator interface for server management, which allows monitoring servers' resource usage and error situations in real time.
REQ-121	10. System administration	Management of load balancers	System must allow using load balancers.
REQ-122	10. System administration	Increased system performance	System must allow scaling system servers both automatically, based on load, and manually. Scaling must not interrupt system operation for more than 15 minutes.
REQ-123	10. System administration	System configuration change log	System must log configuration changes related to memory, hard disc space and number of processors.
REQ-124	10. System administration	Data backup	System data must allow making backups, which can be used to restore data.

ID	Functionality group	Functionality (requirement)	Requirement description
REQ-125	10. System administration	System configuration backup	Backing up configuration of system servers must allow using the backup copy to automatically restore the system. Setting up a clean server can be solved by other automated means, e.g. by using configuration repository that stores all configurations.
REQ-126	10. System administration	Administrative interface	System configuration requires an existing administrative interface.
REQ-127	10. System administration	File version management with file locking	System must allow automated creation of file versions so that the previous version of a file is saved when making changes. It must allow locking the file for the time of making changes.
REQ-128	11. Machine learning	Using machine learning algorithms	System must contain machine learning engine, which allows using machine learning algorithms and setting up machine learning workflows. Machine learning engine can be independent module not directly related to analytics module.

Table 7. Functional and non-functional requirements for big data system

2 Data in big data system

Big data system must contain data that meet the following criteria:

1) Data are necessary for creating and/or providing big data system services, but they do not exist in other databases or there are no relevant query services or such services do not have sufficient availability.
2) Data correspond to the quality standard of big data system.
3) There are no legal restrictions on using data in big data system and legal basis has been established for using the data, i.e. database of big data system has been established.

If there are existing web services for using data and the provision of services does not require copying the data in big data system due to system stability or performance, then these data will not be transferred to the big data system database.

2.1 Databases used by big data system

At the time of preparing this analysis, big data system is known to include the following datasets:

Dataset code	Dataset	Comment
AMA	Agrometeorology data	There is no official database. In the course of the project, we compiled a definition of metrics, which will clarify specific data composition.
TMA	Plant pests monitoring data	Currently, no official database has been created for storing these data. In the course of the project, we compiled a definition of metrics, which will clarify specific data composition.
EMÜ	Data from various databases of EULS that currently lack management software.	No official database has been established. In the course of the project, we compiled a definition of metrics, which will clarify specific data composition.
PANDA	Digital database of agrochemical parameters of Estonian field soils.	There is currently no official database. In the course of the project, we compiled a definition of metrics, which will clarify specific data composition with regard to soil analyses. Another option regarding PANDA data may be interface with ARC LIS. LIS or ARC laboratory information system is currently in development.
TA	Data from Road Weather Stations	There is currently no official database.
Metadata. including classifications	Definitions of used data.	Metadata structure is defined in a separate chapter. Metadata are divided into definitions of metrics and dimensions and data models.

Table 8. Databases included in big data system

Data used and mediated by big data system but not stored in the database of big data system include:

Dataset code	Dataset	Comment
CCS	Cross compliance system	Cross compliance system contains data regarding requirements for producers, which are currently distributed as web articles.

Dataset code	Dataset	Comment
CLIDATA	Weather data collected by Estonian Weather Service.	It is planned to use weather data from different sources to compose a uniform service that would issue data from all sources. This means that for performance reasons, some CLIDATA data might be copied to big data system database.
ESTHub	Data of indexes calculated based on satellite images.	Big data system does not use the data of satellite images. Agriculture requires e.g. plant mass index data.
ETAK	Estonian National Topographic Database	Data are used as base data in analyses and visualisations related to spatial data.
ETKI	Data of plant variety comparison tests from ECRI and ARC.	The existing database of plant variety comparison tests will be linked to web services that allow using data in third party applications.
FADN	Agricultural accounting database.	FADN data anonymous visualisation layer is created in big data system.
GEOPORTAAL	Estonian Geoportal inspire.maaamet.ee	Uses metadata from Geoportal.
KOTKAS	Environmental decisions infosystem	Uses information on environmental permits.
PORTAAL.AGRI.EE	Portal of Veterinary and Food Board, PMA and Ministry of Rural Affairs, which enables exchange of data in form of notices and documents between agencies and producers.	Creating a big data system user starts in customer portal, where user can register as user of big data system. Portal also used for system-related correspondence.
TAKS	Subsidy administration and control system.	Uses and mediates only subsidy-related data.
PRIA-TUKS	PRIA information system for administration of market organisation measures.	Big data system visualises information on application of market organisation measures.
MATS	Rural development subsidy system.	Mediates only subsidy-related data.
MAIT	Information system of rural development plan investment aid.	Mediates only subsidy-related data.
TSR	National register of food and feed business operators.	Mediates data related to business operators.
TKV	Register of plant protection products.	In the course of the project will be created web services that allow using the data.
MPR	Register of organic farming.	In the course of the project will be created web services that allow using the data.
MATER	Register of undertakings operating in the field of land improvement.	In the course of the project will be created web services that allow using the data.
MSR	Register of land improvement systems.	In the course of the project will be created web services that allow using the data.

Dataset code	Dataset	Comment
TTR	Plant health register	In the course of the project will be created web services that allow using the data.
VAETISEREG	Fertiliser register	In the course of the project will be created web services that allow using the data.
SORDIREG	Plant variety register	In the course of the project will be created web services that allow using the data.
MIS	Land cadastre	Cadastral unit data are used.
SMKA	Large-scale soil map with various soil parameter data.	Possible to use with XGIS map application by Land Board

Table 9. Databases used by big data system

3 Architecture variants

Main arguments when choosing the architecture arise from user needs and data properties such as:

1) data complexity;
2) total volume of data and volume of data processed within one service;
3) speed of increase in data volume;
4) requirements for keeping data up to date;
5) performance requirements for application.

Figure 6. Statistics of issues arising in analytics projects (source: Gartner, Inc, 2018)

Diagram shows that main issues in big data and analytics projects are more related to organisation and processes than technologies. Thus, there is no reason to attribute excess importance to technological choices or make it an aim in itself. IT market has plenty of well operating technologies without clear advantage compared to others. The question is rather in the content of data, their interoperability or whether different data sources use analogous data structures, whether classifications are compatible and whether data quality is sufficient to organise cross-usage.

Below are described various aspects that need to be considered when choosing the architecture.

3.1 Collecting vs connecting

One decisive point when choosing the architecture consists in finding balance between data collection and connecting to databases. As a rule, excessive data collection leads to a situation where collected data are not updated fast enough. This leads to reduced quality in terms of data being up to date and accurate. It is reasonable to collect only the data that are not collected by anyone else or that are not sufficiently available from original source.

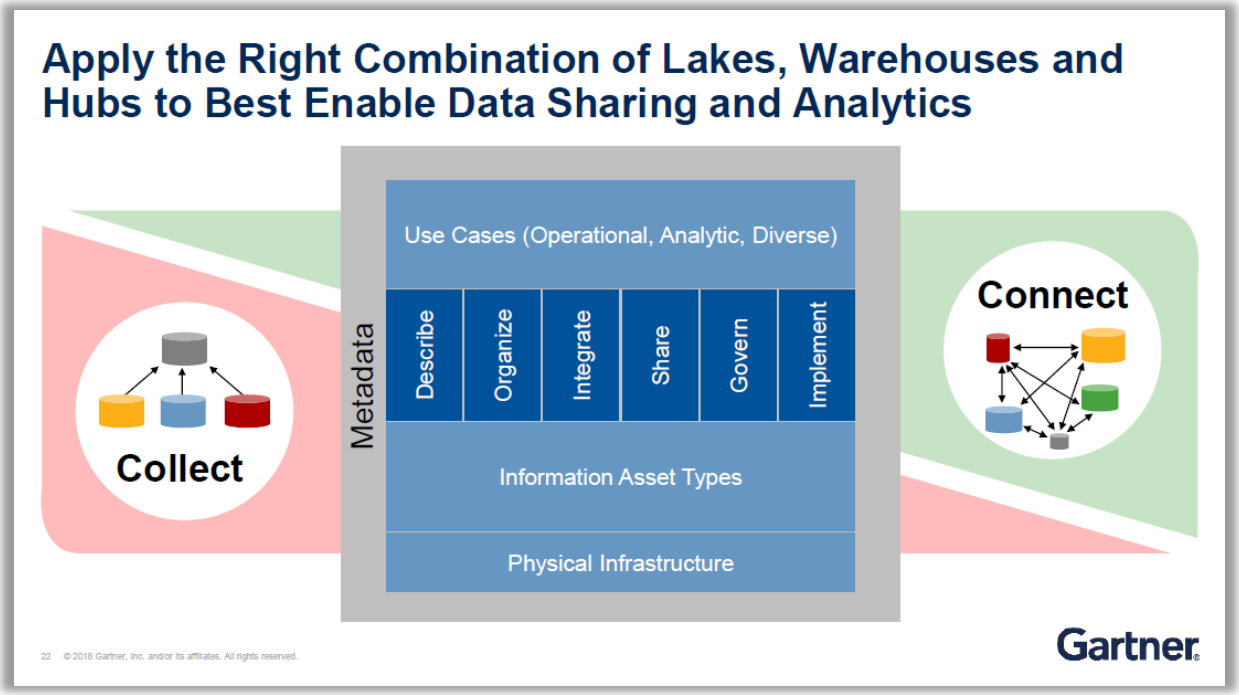

Figure 7. Finding balance between collecting and connecting (source: Gartner, Inc, 2018)

Data cross-usage works though services and through combining data in a single central database. Both variants have their pros and cons.

Pros for data collection:

1) data are quickly accessible;
2) data can be made available after processing and in the most suitable format;
3) data quality can be comprehensively assessed;

Pros for connecting to databases:

1) data are up to date;
2) moving only the data that need to be used;
3) server resources are not wasted on data volume and processing, the results of which may not be used in future;

Cons for collecting concern mostly the amount of time and resources spent on collecting and risk that when collecting data, the collecting process might have gone out of control and data storage may have set of data that differs from the source data.

As for connecting, the disadvantage consists in slow operation when large amount of data is involved. Large-scale analyses can be very slow when connecting to databases, because data query from the source takes place over relatively slow connection. Ultimately, this may mean that the analysis will have no results.

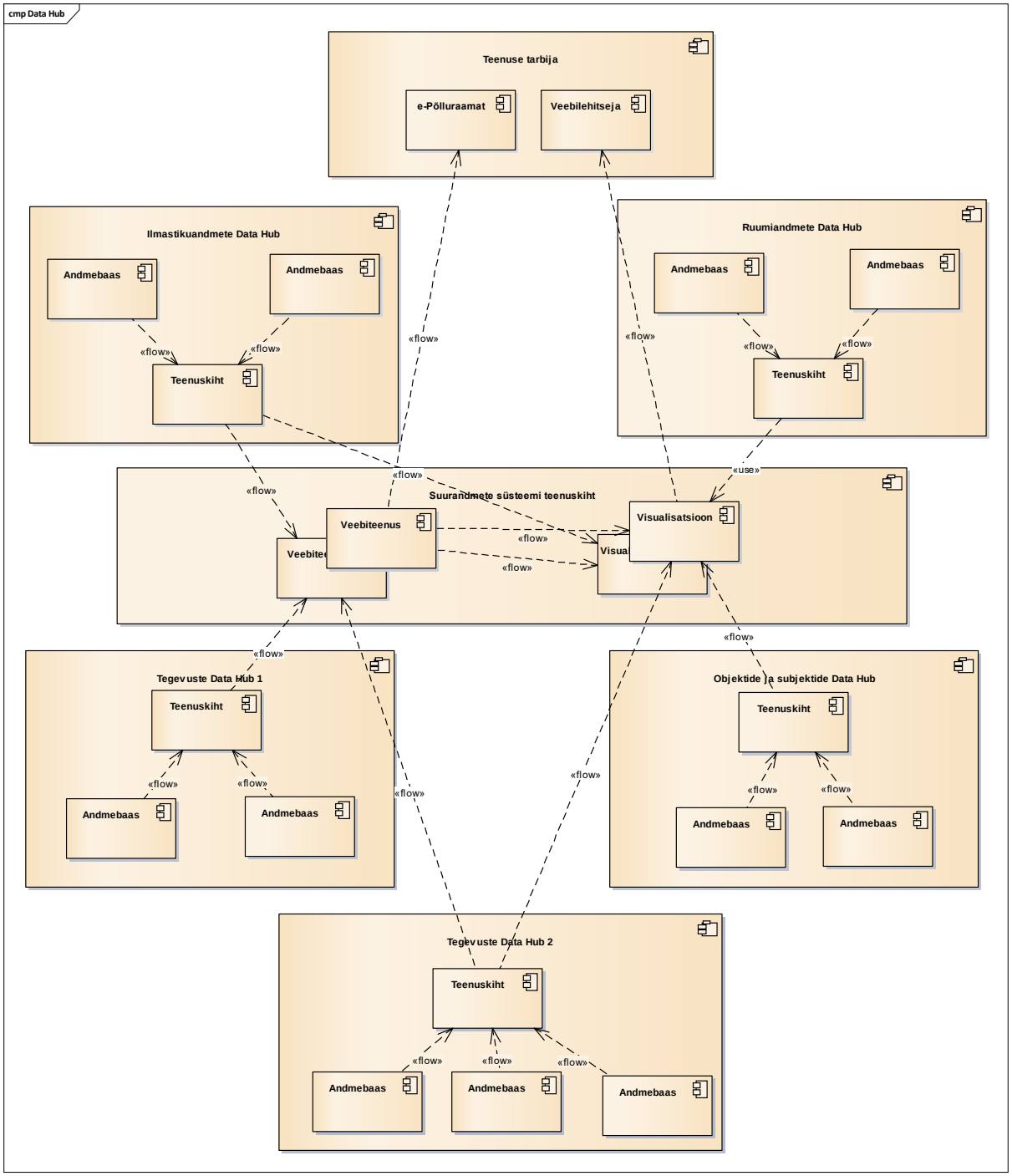

Figure 8. Connecting based architecture

Recently, the term *Data Hub* has been introduced, which symbolises the architecture based on so-called data centres. Data centre contains a series of databases of sectors with similar content and web services that work based on uniform standards.

In Estonia, one example of *Data Hub* consists in the databases and services of the Land Board. They are described in common metadata system http://inspire.maaamet.ee/.

Another system that can be viewed as *Data Hub* is PMAIS, which combines several PMA registers, whereas all registers have their own statutes and are legally considered separate databases.

As a rule, data exchange with *Data Hub* takes place via services, not directly between databases. *Data Hub* may contain miscellaneous connections, e.g.:

1) REST web services (e.g. MATS).
2) File transfer.
3) X-Road services

When viewed from outside, *Data Hub* should constitute a single clearly delineated whole.

Connecting or using data according to particular need is a more vital variant when data to be used should be always updated and they are not used in large amounts. Connecting can be used if data source availability class is higher or equal to the system using the data. Otherwise it will reduce the availability of data user. Using several data sources with low availability will further decrease the user's availability class to lower level than that of the sources.

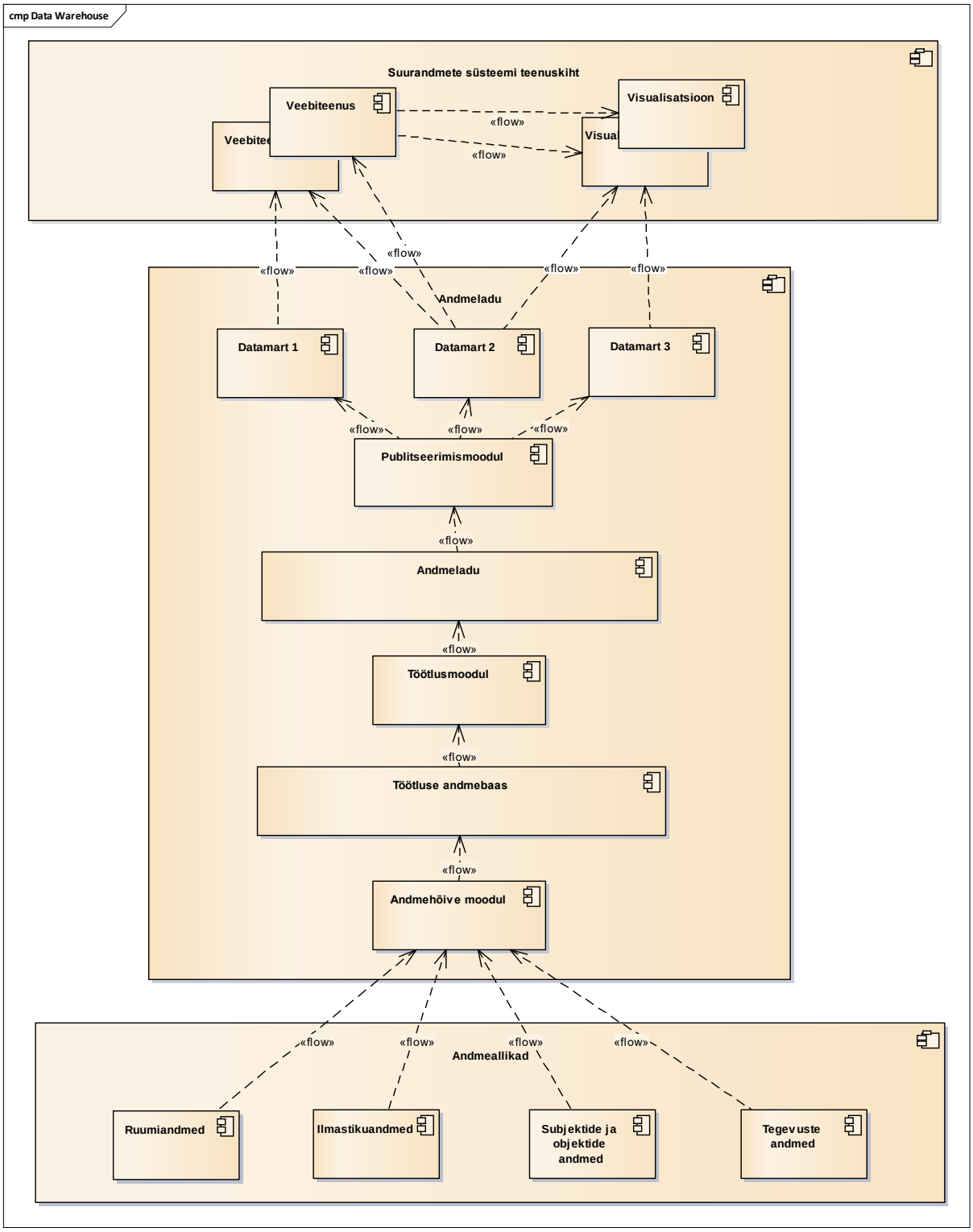

Figure 9. Collecting based system architecture (*Data Warehouse*)

Figure shows typical three-level data warehouse, which, in terms of databases, consists of processing database, data warehouse and so-called *data marts* or analysis databases.

Advantage of collecting consists in quick availability of data. However, this speed is deceptive, because data at the source have already changed when we start using them from analysis database.

In the course of database analysis, it was ascertained that there are several technically advanced data centres established in Estonia. According to recent widespread approach, the data of other database were copied to own database to reduce external dependency. But such solution does not work in case of big data, because many data change so quickly that it causes integrity problems during data transformation and copying, i.e. there is no way to trace the progress of data transfer, because the surrounding data have already changed. Moreover, it is impossible to copy all data within one transaction, because such query would take a long time to work and would most likely end with filling up server memory. There is no point in copying to local system all such data that have functional service for data sharing. This does not mean that it is prohibited to copy data in a document where the data from interfaced database are used. It still has to be done, but it is important to maintain current status of data to be able to determine the data underlying the decisions later on.

In the course of big data system analysis and elsewhere in the world people have come to conclusion that building large-scale data warehouses might not be a functional solution. Data volume, complexity of structure and speed of changes does not allow creating data warehouse that works in real time. Thus, it is necessary to find a balance between connecting and collecting. One should collect only the data that cannot be retrieved from databases in real time. Here is one exception – large-scale analyses that require using entire dataset at the same time and/or repeatedly. In case of the latter, copying of entire dataset to analysis environment is justified.

Therefore, prerequisite for successful creation of this big data system is finding balance between collecting data and connecting to data. It is inevitable to take into account already created data centres such as ESTHub, Inspire, PMAIS and PRIA's register of subsidies and fields. It is not reasonable to continue duplicating these data in full extent.

In case of big data system, it is inevitable to combine certain data in data warehouse. They include agricultural data that are significant but currently have no database. These data should be legalised as data in big data system.

3.2 Cloud vs local installation

There are two principally different methods of installation:

1) in cloud;
2) local installation.

Third possible variant is private cloud that combines certain advantages of cloud and local installation.

Property	Cloud	Local	Comment
Speed of installation	+	-	Speed of installation depends on the level of automation of installation. As installations are constantly repeated in cloud, the installation process is better established and allows automation.
Cost of installation	+	-	There is no fee for installing virtual servers in cloud. There is also not need to purchase licences or hardware. Later on, you will have to pay for services.
Environment stability	+	-	Cloud stability on infrastructure level is generally better than that of local installation because of using virtual machines.

Property	Cloud	Local	Comment
Security	+	-	Cloud offers better security because attacks are monitored on a wider scale and security systems are more advanced.
Monitoring	+	-	Well-known and large clouds have developed very detailed monitoring solutions for both service health and resource utilisation.
Monitoring of resource cost	+	-	For cloud, the basis for calculation consists in server resource used. Therefore, cloud service invoices indicate resource use very accurately and in great detail.
Cost at maximum resource utilisation	-	+	Advantage of local installation is better financial efficiency, as upon full utilisation of server resource, the increase in costs is relatively low. In cloud, the cost of service depends on the space and number of processor cores used.
Ease of scaling up and down	+	-	In a cloud, extending or reducing system hard drive and processors takes just one click or is fully automated. This may be more complex in case of local installation.
Cost of scaling	+	-	Scaling itself does not cost anything in cloud. You only have to pay for using the resource. In case of local installation, immediate investment is required in case of extension scaling. Scaling in cloud is especially efficient in case of constantly changing load. As invoices are issued at a great level of detail, it is possible to divide the cost between consumers, indicating specific bases for cost calculation.
Administrative costs	+	-	Invoice for maintaining a system with reduced load is almost non-existent. The only cost comes from using the resource.
Control over data	-	+	Local installation allows better control over data than in cloud. However, local environment is at greater risk for problems because it usually has less advanced security measures.
Presence of installation helper and functional adviser	+	-	More widespread Azure and Google clouds have expert system that provides server administrator with constant assistance, including in issues related to the security of system installation.

Table 10. Advantages and disadvantages of cloud and local installation

If it has a well-developed infrastructure and administrators, a private cloud or cloud based on local infrastructure is almost as good as large commercial cloud. Another option is to use commercial cloud when launching the system in order to avoid major investment without knowing how quickly the system will become functional. Moving to private cloud may be considered after reaching critical resource usage to reduce resource usage costs.

3.3 Basic software with or without support

Basic software cannot be used without support. Support manifests in constant improvement of software functionality and patching security breaches. National information systems cannot operate without support for security reasons. This means using own support team to ensure constant development of

the basic software or use supported software. One possibility to ensure support is to use software with large user community which facilitates getting required support.

3.4 Closed source vs open source

It is not recommended to use base software with fully closed source. Open source must be available at least with regard to interfaces and adapters to allow quick intervention, repair or supplementing of interface in case of potential problems with data transfer and avoid blocking the system for a longer period.

3.5 Data exchange with databases – web services, data base connections, files

State information system consists of large number of databases that are defined to lesser or greater extent in riha.ee portal. In the course of this project we analysed 41 state databases. X-Road functions as national data transfer channel. It suits well when transferring small volume of data regarding one person or company. Data transfer by using X-Road becomes inefficient in case of moving large volume of data in real time.

As for open data, state databases should use RESTful web services, more specifically, services based on Open Data standard, implementing of which has been described in greater detail in the standards chapter of this document. At the moment, there are very few databases in Estonia that have already created such services (RESTful). Excellent examples are certain datasets of Land Board and Statistics Estonia.

Here it is important to note that in future, we can use the following data sharing variants:

1) each database creates own data sharing services and big data system uses them to provide own synthesising services.
2) big data system will provide both data sharing and synthesising services.
3) data sharing service will be provided by Statistics Estonia, who acts in accordance with new Statistics Act. Big data system receives data via Statistics Estonia and provides own service based on these data.

Most likely, the best variant here is to find middle ground. This means that Statistics Estonia could start providing data sharing service regarding the data that are not yet used by big data system services, and big data system would use services established to particular database to provide own services that give value-added. This requires making a strategic decision whether big data system will share only data enriched by itself or will it also share individual data from various databases without improving them or adding synthesised data. From the aspect of economic viability, it would not be advisable to have big data system providing data sharing service with the same format and data composition as Statistics Estonia. Big data system might provide the service itself in case the form or content of data sharing is not the same as those provided by data sharing service of Statistics Estonia.

3.6 Data transfer with agricultural machines

Analysis revealed that one considerable problem for producers consisted in data exchange with agricultural machines. Different machine manufacturers have developed own systems. These systems are not compatible. As a rule, the systems also offer full functionality for a certain fee. This means that when a producer has n different makes of machines, he would generally have to use n software and pay n contract fee. As a rule, it is not possible to use the same control panel for machines made by different manufacturers, because they are closely related to the software created by machine manufacturer.

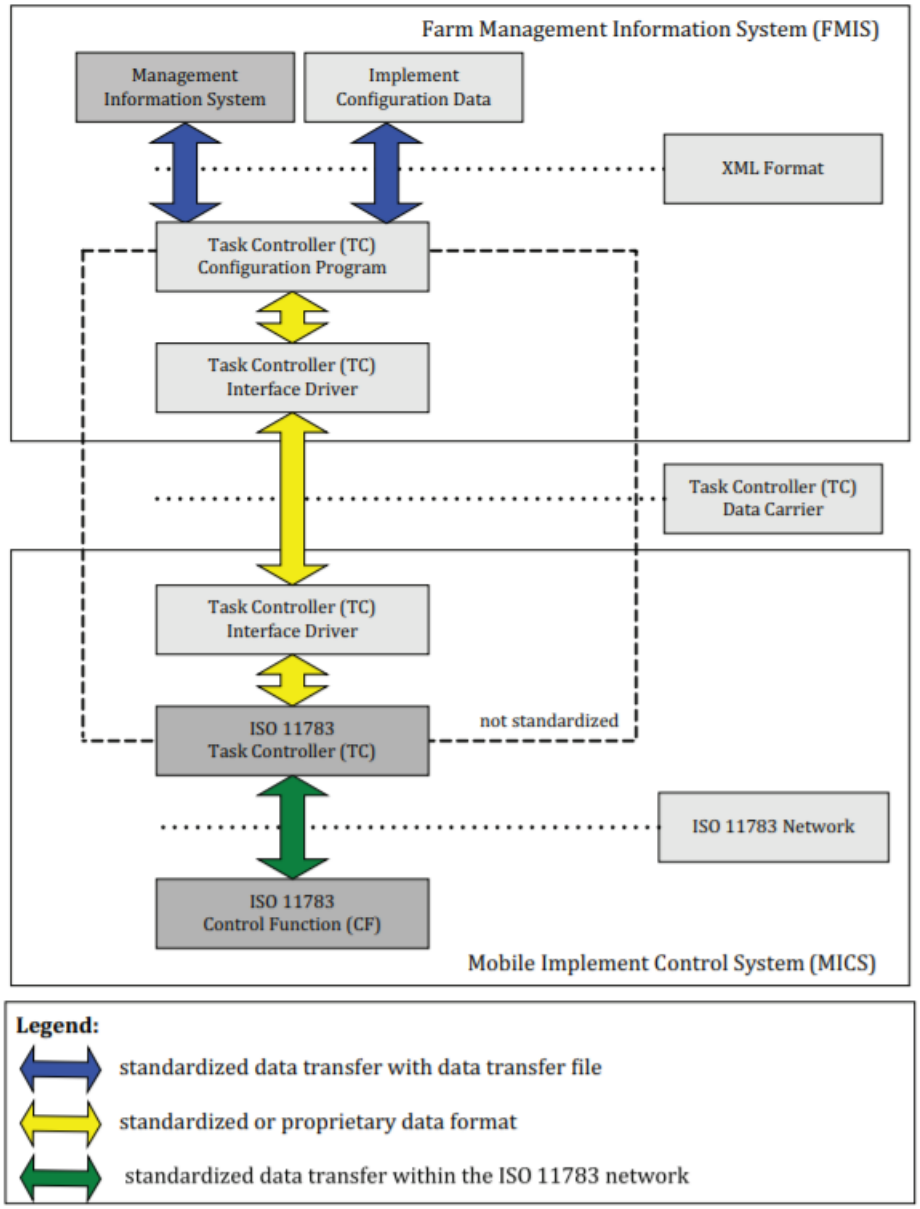

Figure 10. Data transfer diagram based on Standard ISO 11783 (source: ISO)

The diagram shown in the figure is describes data transfer between machine and farm information system (FIM). Data transfer is two-directional. Machine or device is given instruction data from the farm system, e.g. fertilisation map. By using the map and GPS signal, the machine will control various aggregates, e.g. opens and closes nozzles for spraying fertiliser or plant protection product.

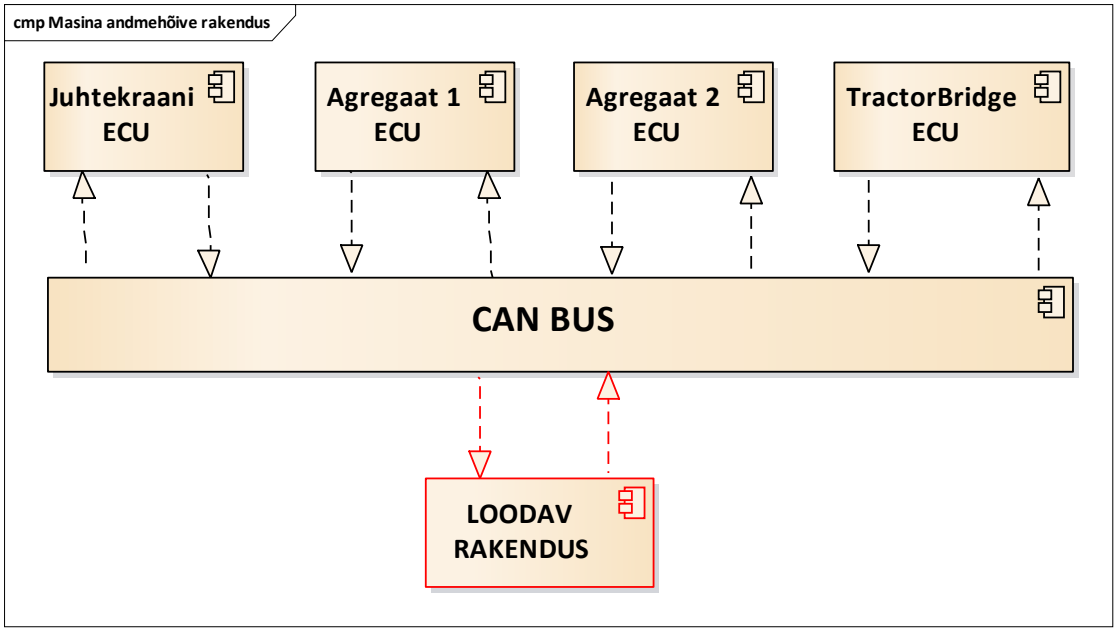

Figure 11. Architecture of data transfer application interface of an agricultural machine

Based on the aforesaid, we concluded that a software for exchanging data with machines must be created in the course of big data project in order to ensure uniform data transfer with all machines that meet the requirements of ISO 11783 standard family. This concerns primarily reading machine operational data and storing data in electronic field data book. When using data, one should keep in mind that their quality may vary with different machines. Before storing data in electronic field data book, data need to be cleaned where necessary. Even if the data must be cleaned, automated data transfer with the machine would mean considerable time saving when entering data in field data book.

It is also necessary to consider that each machine manufacturer has its own rules (machine use agreement), setting limits to allowed data use. On the one hand, restrictions are related to machine safety and reliability, and on the other hand, also to machine manufacturer's copyrights.

ISO standard describes particular data structure used for exchanging data. The following represents operational data collected during tractor test drive in the course of the project, in the format described in ISO standard:

```
<?xml version="1.0" encoding="UTF-8"?>
<ISO11783_TaskData VersionMajor="2" VersionMinor="0"
ManagementSoftwareManufacturer="AgLeader Technology" ManagementSoftwareVersion="4.0_7537"
TaskControllerManufacturer="AgLeader Technology" TaskControllerVersion="4.0_7537"
DataTransferOrigin="2">
  <TSK A="TSK-0" B="2018-12-10_12:08:43" G="4" C="CTR-0" D="FRM-0" E="PFD-0">
    <TIM D="4" B="2018-12-10T12:27:13">
      <DLV A="0051" B="896" C="DET-1"/>
      <DLV A="0051" B="896" C="DET-1"/>
      <DLV A="0051" B="896" C="DET-1"/>
      <DLV A="0051" B="896" C="DET-1"/>
      <DLV A="0051" B="896" C="DET-1"/>
    </TIM>
    <DAN A="A00A80000B202AB4" C="DVC-0"/>
    <PAN A="PDT-0" E="DET-1">
      <ASP D="4" A="2018-12-10T12:09:58"/>
```

```
    </PAN>
    <PAN A="PDT-0" E="DET-1">
      <ASP D="4" A="2018-12-10T12:12:13"/>
    </PAN>
    <PAN A="PDT-0" E="DET-1">
      <ASP D="4" A="2018-12-10T12:17:13"/>
    </PAN>
    <PAN A="PDT-0" E="DET-1">
      <ASP D="4" A="2018-12-10T12:22:13"/>
    </PAN>
    <PAN A="PDT-0" E="DET-1">
      <ASP D="4" A="2018-12-10T12:27:13"/>
    </PAN>
    <TLG A="TLG00000" C="1"/>
  </TSK>
  <PFD A="PFD-0" C="KOMMU" D="132277" E="CTR-0" F="FRM-0">
    <PLN A="1">
      <LSG A="1">
        <PNT A="2" C="58.8034226" D="26.7522538"/>
        <PNT A="2" C="58.8033372" D="26.7523559"/>
        <PNT A="2" C="58.8031837" D="26.75251"/>
        <PNT A="2" C="58.803341" D="26.7533593"/>
        <PNT A="2" C="58.8033448" D="26.753394"/>
        <PNT A="2" C="58.803337" D="26.7534106"/>
        <PNT A="2" C="58.8033237" D="26.7534067"/>
      </LSG>
      <LSG A="2">
        <PNT A="2" C="58.8035278" D="26.7559649"/>
        <PNT A="2" C="58.8035322" D="26.7560398"/>
        <PNT A="2" C="58.8035173" D="26.7561103"/>
        <PNT A="2" C="58.8034909" D="26.7561598"/>
        <PNT A="2" C="58.8034562" D="26.7561828"/>
      </LSG>
      <LSG A="2">
        <PNT A="2" C="58.804132" D="26.7551101"/>
        <PNT A="2" C="58.804149" D="26.7552204"/>
        <PNT A="2" C="58.8041426" D="26.7553087"/>
      </LSG>
    </PLN>
  </PFD>
  <FRM A="FRM-0" B="TO" I="CTR-0"/>
  <CTR A="CTR-0" B="KU"/>
  <DVC A="DVC-0" B="New Holland T7.2, Kverneland EDW" D="A00A80000B202AB4"
F="00000000000000" G="FF000000000000">
    <DET A="DET-0" B="1" C="1" E="0" F="0"/>
    <DET A="DET-1" B="2" C="2" E="1" F="1">
      <DOR A="3"/>
      <DOR A="4"/>
    </DET>
    <DET A="DET-2" B="5" C="4" E="2" F="2">
      <DOR A="6"/>
      <DOR A="7"/>
```

```
    </DET>
    <DET A="DET-3" B="8" C="4" E="3" F="2">
      <DOR A="9"/>
      <DOR A="10"/>
    </DET>
    <DET A="DET-4" B="11" C="4" E="4" F="2">
      <DOR A="12"/>
      <DOR A="13"/>
    </DET>
    <DPT A="3" B="00B3" C="1"/>
    <DPT A="4" B="0043" C="24000"/>
    <DPT A="6" B="0087" C="-11000"/>
    <DPT A="7" B="0043" C="2000"/>
    <DPT A="9" B="0087" C="-9000"/>
    <DPT A="10" B="0043" C="2000"/>
    <DPT A="12" B="0087" C="-7000"/>
    <DPT A="13" B="0043" C="2000"/>
    <DPT A="15" B="0087" C="-5000"/>
    <DPT A="16" B="0043" C="2000"/>
    <DPT A="18" B="0087" C="-3000"/>
    <DPT A="19" B="0043" C="2000"/>
  </DVC>
  <PDT A="PDT-0" B="Amm. Nitrate"/>
</ISO11783_TaskData>
```

More detailed information about ISO standard and various parameters it uses is available on our webpage https://www.isobus.net/.

In order to use ISO-based data, farm infosystem and state electronic field data book must be brought in conformity with the requirements described in ISO 11783 standard.

Alternative variants for exchanging machine data include:

1) creating state level data exchange module that can read data on CAN BUS level.
2) creating adapters for all software used by machine manufacturers.

Both variants have their cons as well. In case of the first one, it is rather difficult to retrieve and process correct data. Disadvantage of the second variant consists in large number of parties and legal obstacles for data usage.

Related activities upon realisation of both variants include contributing by the example of open software created as EU level competence leader and demonstrate its benefits to the society, achieving a situation where certain machine manufacturers' data are made freely available for certain purposes (e.g. automated data collection). By freely available we mainly mean making machine user agreements more reasonable in terms of producer's data usage rights. This project involved compiling a list of data elements (metrics) that should be retrieved from the machines.

Motivation tools must be created on EU level (e.g. incentives as positive motivation or restrictions as negative) for machine manufacturers to attract their interest in granting the right to use data.

Legal instruments for standardising machine data exchange include establishing mandatory usage of ISO 11783 by EU directive, specifying the composition of used data that must be available in standard

format from the machines of every machine manufacturer. Data composition can be based on the abovementioned list.

Pilot project for creating machine data acquisition software would potentially highlight social benefit and thus also support making bolder steps. There is a risk that producers will not be motivated, but there is also an opportunity that someone will be first and grant the right of using data, after which competition forces others to follow. It is important to shift the mentality and respecting agricultural producer's rights to use his own data. Here, it is important to keep an eye on the progress made in NIVA project, partners of which include ARIB from Estonia; the subject of machine data is led by project partner from Netherlands.

4 Technical architecture variants

As for technical architecture we suggest 3 variants:

1) SQL Server 2019 Polybase and Azure and Power BI based architecture.
2) Apache Hadoop and MapReduce based architecture.
3) Spark and PostGIS based architecture.

4.1 Performance of databases

Speaking of relational databases, important aspects with regard to data volume are the number of entries and the number of bytes per entry.

The table below gives an example of performance parameters of three database systems with spatial information processing capacity.

Volume (points)	PostGIS (ms)	Spatial Spark (ms)	Elcano (ms)
1000	234	6,543	9,516
10,000	326	6,622	9,714
100,000	3783	8,301	9,030
1,000,000	29,899	8,301	10,747
10,000,000	269,257	20,487	17,099
100,000,000	5,752,821	55,017	37,378
1,000,000,000	More than 10h	399,100	273,074

Table 11. Performance of different data processing systems when processing data (source [1])

Performance test shows that there is a huge difference in the capacity of systems. Is this difference important for the user of big data system? When looking at the list of services of big data system, the needs that have occurred first are mainly related to the field dimension, such as the service for calculating field humus balance. This service is processed to maximum extent or it has 10,000 data entries. At such a number of entries, the technical solutions intended specifically for processing big data are very slow, exceeding generally recognised critical response time, which is ca 6-7 seconds. Therefore, when creating field-specific services, it would be erroneous to use big data technology.

It is reasonable to use big data processing technology when bulky (in this case 1,000,000,000 entries) data analyses are performed more than ca 120 times a day. Currently, there are no known consumer services that would process so many data. Thus, application of big data technology to consumer services has a questionable value.

The results of performance tests indicate that the resource need of currently known services does not exceed usual database software resource need. Therefore, Postgre and PostGIS software is generally suitable for processing data in terms of database.

However, big data processing technology may be useful when preparing services, e.g. in research institutions that run analyses involving entire Estonian dataset. The volume of one map layer measured by LIDAR may reach 45,227,000,000 points when the level of detail is 1x1 metre. In case of so large data volumes, the usual Postgre database can no longer be used in real time. Processing such a volume requires a separate environment with different architecture.

4.2 Horizontal scalability of the solution

There are different variants for solution scaling. When using cloud service, it is rather easy to increase system performance by increasing the number of processors, as well as disc space used. After reaching its limits, the next step would be sectioning the solution. With regard to all offered architecture variants, it is possible to create services as different applications, be it then data visualisation or web service for making calculations. These solutions do not have to run on one server or even in one cloud. It is possible to utilise free cloud services (e.g. Power BI) and move on to others when the resource needs increases.

The costs of using Azure and Google clouds increase as the number of users increases, and in this case, it is possible to move most used applications to private cloud, if there is sufficient capacity for local infrastructure management.

The most used data structure of the system in terms of parameters and events is planned in a manner that easily allows transforming data into several databases and merge them back together where necessary. In terms of planned integration tools, it does not matter whether data come from one or several databases. Thus, it is possible to distribute queries to n servers without using advanced technologies. In this case, n stands for the number of event types.

The only thing that needs to be panned for one database server is metadata management, which runs the functioning of entire solution.

4.3 Architecture variant no. 1

Microsoft has developed a leading data management and analytics family, which includes:

1) **MS SQL Server 2019 Polybase** – software that uses common SQL language and ODBC and JDBC interfaces to allow using data ini SQL Server itself as well as in other databases, including HDFS (Hadoop File System), Oracle, Teradata and Azure Data Lake Store. Database is equipped with GIS data functionality in the form of both data types (*geometry, geography*) and extensive functionality. SQL Server includes a particularly important functionality in terms of analytic applications – *in-memory* OLTP transactions and *columnstore* indexes, which allow increasing the query performance tenfold. The technology is based on storing data in operating memory and quick retrieval from the memory. In case of SQL Server, it is possible to use clustering (*Always On Failover Cluster*), which means uninterrupted solution, where cluster combines mutually duplicating servers.
2) **SSMS** – SQL Server Management Studio, which allows managing SQL Server database, models and languages for creating and changing a database.
3) **SQL Server Analysis Services** – data analytics engine that contains various software products for reporting and analyses.
4) **SSIS** – SQL Server Integration Services represents data integration and transformation platform with options to connect to files (csv, xlsx) and different relational databases.
5) **Power BI Pro** – a powerful tool for data integration, analysis, making analytic applications, comes with free cloud service that allows integrating data from both relational databases Postgre, Oracle, SQL Server, etc., and web services, Spark API and HDFS. Power BI has integrated R and Python support, which allows more complex data analyses and processing without setting up additional environment. A separate server may be used for R in case greater performance is needed.
6) **Sharepoint** – document and list management, version management.
7) **Teams** – groupwork with documents (xlss, docx, pptx).

8) **Azure** – cloud service managed by Microsoft. The service allows using both Postgres and SQL Server databases, application servers for Java and C# applications, and provides options for load balancing, monitoring and securing entire infrastructure on a very wide scale.
9) **Windows Server** – operating system.

Power BI, SQL Server and Azure cloud service also serve as a basis for the result of the database analysis performed during the project: tietoanalytics.ee/PRIA.

Pros of the solution:

1) Functional two-factor authorisation and authentication system, which can be linked to local domain (*Active Directory*). Same users both in cloud and local installations.
2) Automated scaling (if enabled in cloud).
3) Manual scaling by just a click in case of cloud installation.
4) Solid and proven integration of different tools of the same family.
5) Easy product installation due to software packaged in installation kits.
6) Packages are supported in cloud service.
7) Ongoing and fast software development. Automatic updates in servers and workplace software along with notification system.
8) Data adapters use Power BI with open source.
9) Both adapter services and visualisation components can be developed by the user.
10) Options to visualise both spatial and alphanumeric data.
11) Power BI REST API allows using data of compiled visualisations in other systems.
12) Specialists present in Estonia.

Cons:

1) Basic software with closed source.
2) Some software options have paid licenses (SQL Server and Windows Server).
3) Fast software development forces development of services built on the platform.

4.4 Architecture variant no. 2

One of the architecture variants involves using the developments of Apache projects:

1) **Apache Hadoop** – open source library, developed for managing and analysing large-scale data in order to allow working on thousands of computers simultaneously.
2) **HDFS** – Hadoop Distributed File System is a file system that allows data clustering and offers an opportunity for very fast queries, including aggregation.
3) **Apache Yarn** – application for managing and monitoring calculation capacity.
4) **Appache MapReduce** – allows creating data processing (parallel) processes that cannot be realised with Hadoop tools alone.
5) **Appache Zeppelin** – data visualisation application.
6) **Linux** – operating system.

Pros:

1) Open source.
2) Free licenses.
3) Uniform system can run on one or thousand machines simultaneously.
4) Automatic troubleshooting. Faulty hardware can be replaced without interruptions in system operation.
5) Horizontal scaling of the system without interruptions.

6) Zeppelin REST API allows using data of compiled visualisations in other systems.

Cons:

1) Free versions have no official support and one has to arrange support by oneself.
2) There is no automated distribution system for software updates.
3) MapReduce does not allow *in-memory* data processing, which makes it significantly slower than many other parallel processing solutions.
4) Majority of work is performed on the command line. There is no graphic user interface for users without programming skills to analyse the data.
5) Free versions do not have established packaged installation set system, which makes it very difficult and time-consuming to search for and find necessary components.
6) There is no default option to link authentication and authorisation to local LDAP.
7) By today, Hadoop is old technology and does not keep up with new big data handling products offered in cloud. Clouds have basically realised the default Hadoop advantages.
8) There is no adequate network of specialists in Estonia who would be able to install and manage the system.

4.5 Architecture variant no. 3

Alternative architecture for Hadoop and MapReduce:

1) **Apache Spark** – parallel processing framework, which allows *in-memory* processing and analyses. As it allows both *in-memory* data usage and parallel processing simultaneously the performance indicators are remarkably high.
2) **GeoSpark** – Spark extension for performing spatial information analysis.
3) **GeoSparkViz** – Spark extension that allows visualisation of spatial data.
4) **Appache Zeppelin** – data visualisation application.
5) **Postgre** – commonly used database engine in the domain of the Ministry of Rural Affairs, using of which should be still considered for storing and processing relational data in big data system.
6) **PostGIS** – extension for Postgre database, which allows operations with spatial data. PostGIS does not include automated clustering. You have to create clusters yourself, by dividing data into sub-databases that contain datasets that need to be processed at the same time. This allows efficient utilisation of *in-memory* options in case of Spark.
7) **MongoDB** – for storing data in form of XML and JSON documents.
8) **Pgadmin** – Postgre database management software.
9) **Linux** – operating system.

Pros:

1) Open source.
2) Free licenses.
3) Superfast data processing and analysis.
4) Options for visualising spatial information and alphanumeric information.
5) Zeppelin REST API allows using the data of compiled visualisations in other systems.

Cons:

1) Free versions have no official support and one has to arrange support by oneself.
2) There is no automated distribution system for software updates.
3) MapReduce does not allow *in-memory* data processing, which makes it significantly slower than many other parallel processing solutions.

4) Majority of work is performed on the command line. There is no graphic user interface for users without programming skills to analyse the data.
5) Free versions do not have established packaged installation set system, which makes it very difficult and time-consuming to search for and find necessary components.
6) There is no default option to link authentication and authorisation to local LDAP.
7) There is no adequate network of specialists in Estonia who would be able to install and manage the system.

4.6 Choosing the architecture

Functionality	Variant 1	Variant 2	Variant 3
Visualisation and publication of data	• Power BI Pro • Power BI Cloud • Azure Cloud	• Apache Zeppelin	• Appache Zeppelin • GeoSparkViz
Analysis	• Power BI Pro • SQL Server Analysis Services • R • Python	• Apache Zeppelin • Apache MapReduce • R • Python	• Appache Zeppelin • Apache Spark • GeoSpark • R • Python
Data processing	• SSIS • MS SQL Server 2019 Polybase • SSMS	• Apache Hadoop • Apache MapReduce	• Apache Spark
Data entry and management	• Data entry and management module (special development) • MS Sharepoint • MS Teams • Office 365	• Data entry and management module (special development)	• Data entry and management module (special development)
Data storage	• MS SQL Server 2019 Polybase	• Apache Hadoop, HDFS • HBase	• Postgre • PostGIS
Application hosting and administration	• Windows Server • Azure Cloud • Power BI cloud • (or private cloud)	• Linux • Apache Yarn • Google Cloud (or private cloud)	• Linux • Pgadmin • Google Cloud (or private cloud)

Table 12. Configurations of architecture variants

The analysis revealed that data volumes that the big data system must handle, are small in the meaning of big data definitions. This cannot be said about data complexity or data quality issues. Therefore, the first requirement for architecture consists in the ability to handle complexity and the speed of handling complexity, i.e. the speed of reaching results when creating services.

In those terms, the most complete architecture variant is **Variant 1**, both in case of local installation and cloud installation. What are the grounds for this statement?

Survey company Gartner Inc carries out annual surveys on the products of analytics software manufacturers. Assessment to software is provided in the form of four squares as shown in the figure below:

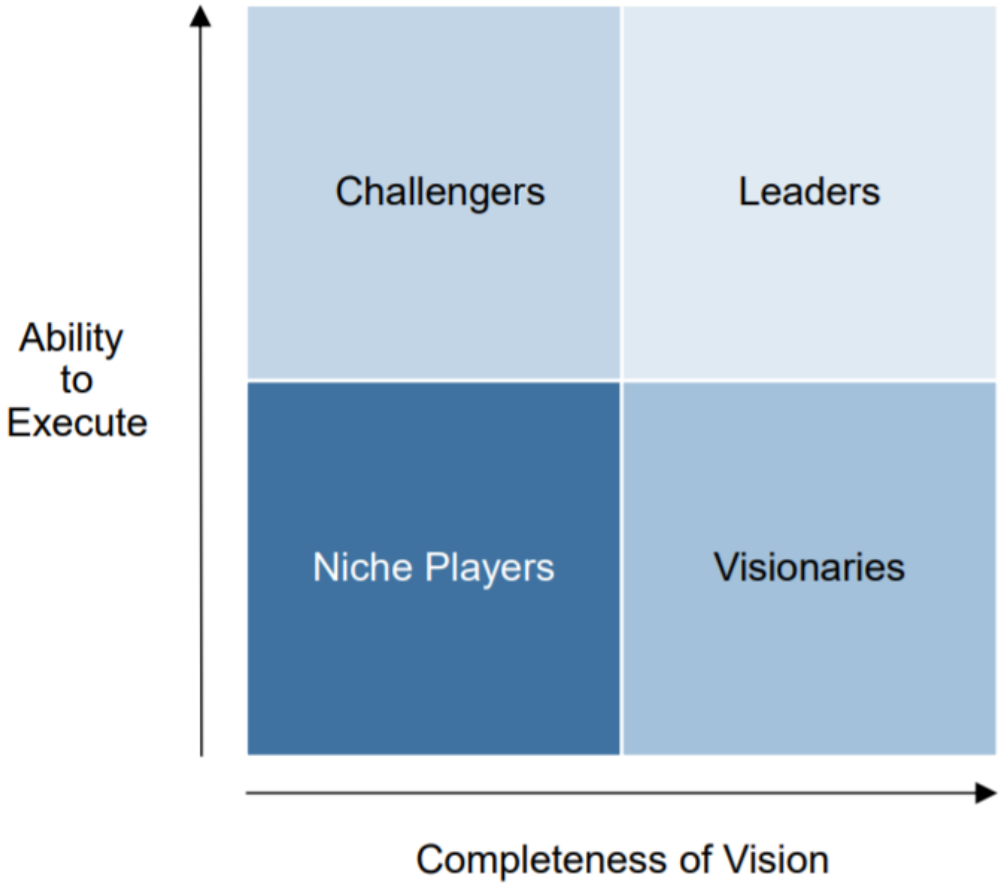

Figure 12. Gartner's Magic Quadrant for assessing software products, Source: Gartner Inc.

Manufacturers are divided into four groups:

1) leaders;
2) visionaries;
3) niche players;
4) challengers.

The horizontal axe shown in the figure indicates completeness of vision in terms of different technologies and solutions and vertical axe indicates the ability to generate applications. Upper right field represents manufacturers whose products allow the user to get the fastest results by using functionality with maximum coverage.

Figure 13. Evaluation of providers of data analytics solutions, source: Gartner Inc.

The figure shows that currently the undeniable leader in terms of both vision and implementation of solution is Microsoft (Power BI, SQL Server, SSIS, SSMS, etc.), followed by Tabeleau and Qlik. Advantages of Microsoft include not just good vision and high efficiency in making applications, but also low costs, which was confirmed in the course of this project. Reduced administration volume was an important component in cost savings.

The first architecture variant was tested in the analysis stage of the project, when creating an analysis application that covers all databases and visualises data quality and content. Positive features included quickly generated results, stability of end user application, ease of administration and low volume of work involved. Project lasted for 11 months and there were no blocking interruptions in the work of the system. Number of users in terms of analysis was 4 and visualisations were used by dozens of people involved in the project.

License cost is often a feared issue for products of Variant 1. In case of given number of users, direct costs for entire system when installed in cloud were ca 1200 euros, which is a very low cost for 11 months, considering that it involved data from 40 databases, tens of users and SQL Server, Azure, Power BI and SSMS software. System management did not need a separate administrative unit consisting of administrators. The first analysis results – interactive analytics applications – were made without developers' assistance and were completed within the first few weeks of the project.

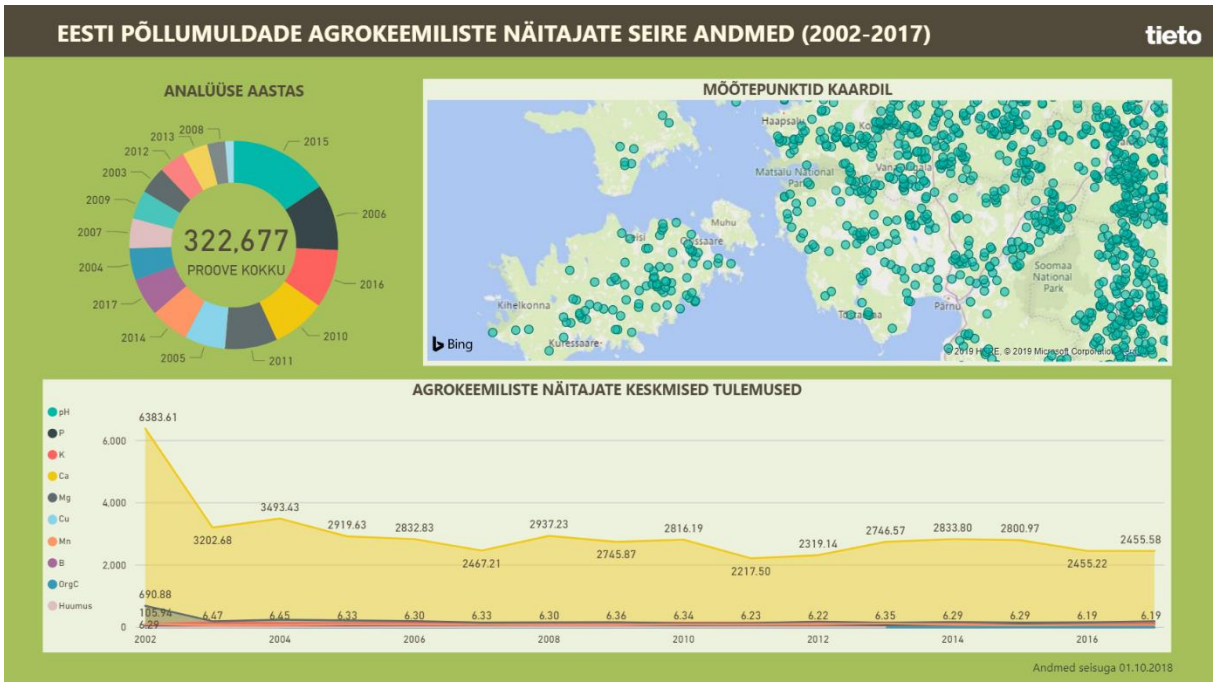

Figure 14. Example of analytics application, summary of agrochemical soil parameters

This application displays soil analysis data in aggregated form.

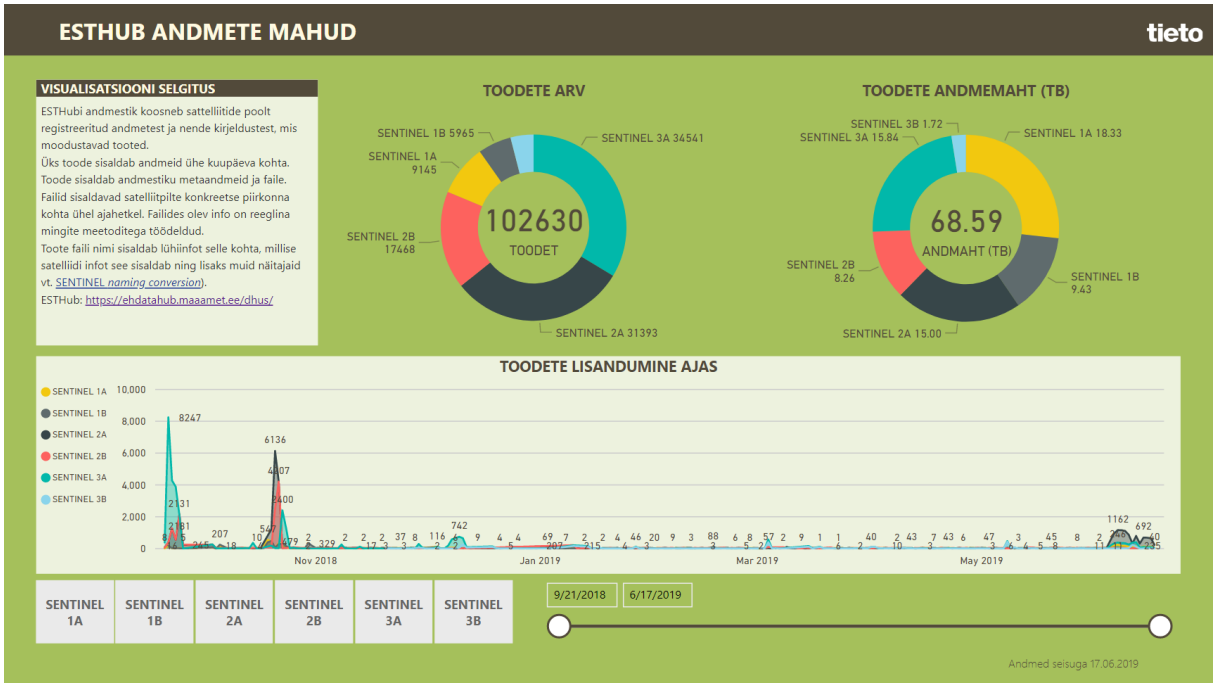

Figure 15. Example of analytics application, monitoring ESTHub data increase

The application shown in the figure above allows monitoring ESTHub data increase based on different dimensions.

Common feature of all mentioned architecture variants is using R and Python for realisation of analysis algorithms. Thus, it is possible to apply programmes created on one platform to other platforms. When focusing on one architecture, it is still possible to make changes to the architecture upon occurrence of additional circumstances, up to the point when it is time to switch to another platform.

5 Description of big data system standards

Big data system standards are defined in order to provide starting points for creating technical interoperability between different databases and data scientific interoperability between related institutions. Without using standards, there would be many one-to-one agreements, and it is difficult to allow interoperability and cooperation between several parties at a time. Using knowledge and data from other database is significantly more difficult unless parties follow the agreements based on established standards. It is also important to have a legal agreement underlying the use of standards.

5.1 Requirements for physical data model

1) Database data model must be documented in a manner that allows machine processing. Most common is UML format as eap. This is supported by tool SPARX Enterprise Architect. This tool is already used for new information systems of the Ministry of Rural Affairs and it should be implemented in other systems as well.

2) Data model must be of UML data model type and it must be possible to transfer it as XMI file (https://www.omg.org/spec/XMI) in open format to other solutions for managing metadata and data models.

3) If up to date physical data model does not exist, it must be created from real database by using reverse engineering, so that the model would describe real physical data model and not someone else's subjective, so-called logical model.

4) Tables must be defined as follows:

    - Name of data table must be in Estonian or in English and express the content of the table as closely as possible.
    - Data table must have comments (Notes), which allows longer description of the contents of the table than the name.

5) Table columns must be defined as follows:

    - Table column must have a name in Estonian or in English that expresses data content of the column.
    - Table column must contain comment (Notes), which describes the data content of the column. If the column is used for storing classification element codes, it is necessary to describe the classification code, elements of which are used in the column.
    - Column data type must be defined among data types available for use by database engine of the database where data are stored. This means also that each table must have defined name of the database engine.
    - Mandatory columns must be marked.

6) Relations between tables must be defined so that it is possible to understand, which field refers to which, and the type of relation (0..n, 0..1, 1..1, n..n).

7) Physical data model must allow generating DDL-script for database creating. It must be possible to recreate entire database at any time without using scripts for implementing incremental changes created during system development.

5.2 Requirements for definition of metrics

Metrics are represented in databases as data elements that describe phenomena in real world from quantitative and qualitative aspects. Metric is one of the foundations of value given by big data system. The value of data with undefined content is questionable in case of precision farming. Despite the fact that some data are undefined, every dataset always comes down to determining their content and using the definition of the content as a basis for subsequent jobs and services. In order to achieve comparability, data must have defined content and data collection methodology. The meaning of data content depends on the device used for measuring or data collecting and on other dimensions used at the time of measurement. Without knowing these particulars, fertilising recommendation, dataset controlling the work of agricultural machine or humus balance might have wrong content. Main task of big data system is to provide future forecast by using data registered in the past. The purpose of compiling all kinds of workplans and forecasts is to achieve desired future and goals by using optimum resources.

Metrics go hand in hand with dimensions, which give them a place in time and space and in additional categories expressed in the form of classifications and code lists. For example, field test metrics are crop and fertiliser quantity and the dimensions are time and place of testing. Objects and subjects related to the test are also treated as dimensions.

Definition of metrics uses the following data elements:

Element	Description
Code	Unique metric code across all metrics defined in the system. As many metrics have similar names, it is reasonable to encode metrics.
Name	Name of metric, which describes the content of the metric. Name metric may give clues to the dimension linked to it, e.g. “Wheat yield”, “Air temperature”, “Soil humidity at the depth of 30cm”.
Definition	Metric definition which gives more specific description of the content of metric.
Measurement unit	Measurement unit used for the metric, e.g. kg, t, s.
Data type	Metric data type. Possible values are “text”, “integer”, “real number”, “classification”.
Format	Metric format, e.g. “xls”, “DDMMYYYY”. This also indicates the number of decimal points of an integer, if it is specified in dataset.
Group	Metric group code. As there are thousands of metrics, it is necessary to group them in order to have better big picture and chances to find accurate metric.
Administrator	Person administering metric definition. This does not refer to the person who generates the metric data or parameters, but the institution or person responsible for managing the metric definition. This also refers to determining the methodology for collecting metric data (parameters).
Sample data	Sample data provide additional description of the metric content.
Data sources	Relation between metric and data sources. This may include several data sources (data source codes).
Is metric original dataset or derivation	Every metric must be defined either as original data or metric calculated based on original data. Possible values include “original data” or “derivation”. If the metric consists in aggregated data, it is always called derivation. Derivation can be generated during calculation from original data, which are subject to not addition operation but a more complex formula.

Table 13. Structure of metric definition

5.3 Requirements for definition of dimensions

Dimension dataset is usually self-explanatory. Therefore, the quality of the dimension is important. However, in order to know which dimensions are used, they should be described in a few words and systemised.

Classifications are essentially dimensions, but not all dimensions are classifications (e.g. dimension of time).

Element	Description
Code	Unique dimension code across all dimensions defined in the system.
Name	Name of dimension that describes dimension content, e.g. "measurement method", "test year", "time".
Description	Brief more detailed description of dimension content.
Group	Dimension group code. As there are thousands of dimensions, it is necessary to group them in order to have better big picture and chances to find accurate one.
Administrator	Persona administering dimension definition.
Classification code	If the dimension represents a classification, this should indicate classification code.

Table 14. Structure of dimension definition

Table describes the data elements used for defining dimension.

5.4 Requirements for classifications

Classifications must comply with the following structure:

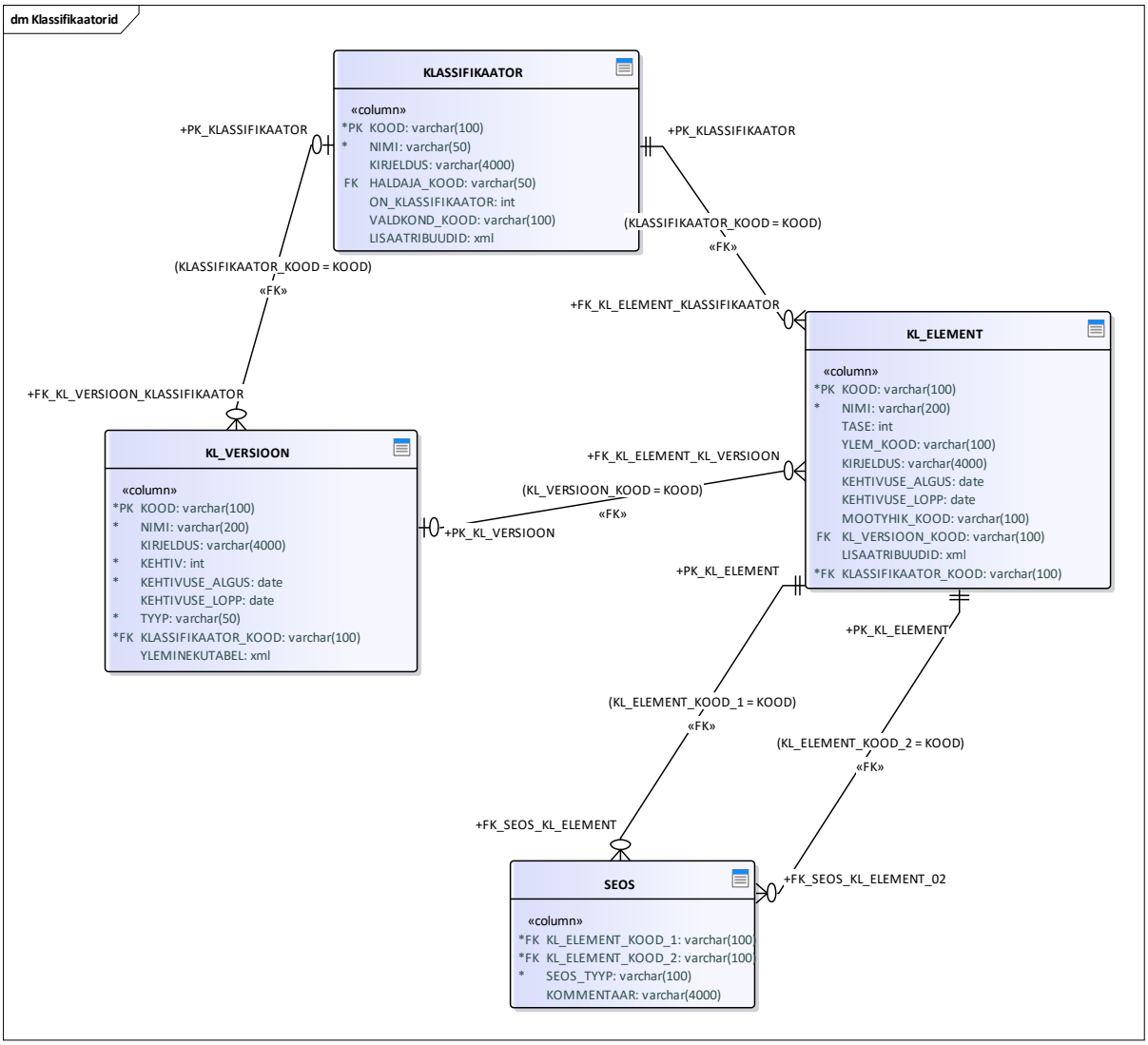

Figure 16. Data structure of classifications

Solution for managing classifications must take into account the following requirements:

1) Classifications management is by nature bulk data processing. It means that classifications and their elements enter and exit the system as bulk data either in CSV or XLSX format. Big data system must have corresponding functionality.
2) System must allow comparison of classifications in order to find recurrent classifications. As a result of the comparison, the system must create relations between classification elements, i.e. record the comparison result.
3) It must be possible to merge and split classifications to get different classifications.
4) Big data system must have web services for classifications data, which allow users to query and, according to their rights, renew classification data.

5.4.1 Table: KLASSIFIKAATOR

Classification used in database. It can be code list, i.e. it does not necessarily cover entire set of data objects in question.

Primary key	Attribute	Default value	Data type	Mandatory	Description
True	KOOD		varchar(100)	True	Classification code or abbreviation.
False	NIMI		varchar(50)	True	Classification name.
False	KIRJELDUS		varchar(4000)	False	Classification definition.
False	HALDAJA_KOOD		varchar(50)	False	Registration number of the institution managing the classification.
False	ON_KLASSIFIKAATOR		int	False	If 1, then it is a classification. If 0, then it is not a classification but code list, which may not cover all objects in question.
False	VALDKOND_KOOD		varchar(100)	False	The domain related to the classification – reference to domain classification element.
False	LISAATRIBUUDID		xml	False	Additional attributes of classification element in the following format: <COLOR>RED</COLOR> <SIZE>BIG</SIZE> <SUBJECT>DATA</SUBJECT>

Table 15. Data structure of classification

5.4.2 Table: KL_ELEMENT

Classification or list element.

Primary key	Attribute	Default value	Data type	Mandatory	Description
True	KOOD		varchar(100)	True	Code of classification element.
False	NIMI		varchar(200)	True	Name of classification element.
False	TASE		int	False	Level indicator of element in case of hierarchical classification.
False	YLEM_KOOD		varchar(100)	False	Code of superior element in case of element at lower levels in hierarchical classification.
False	KIRJELDUS		varchar(4000)	False	Description of classification element, explaining when the object in question belongs to this element.

Primary key	Attribute	Default value	Data type	Mandatory	Description
False	KEHTIVUSE_ALGUS		date	False	Beginning of validity of element.
False	KEHTIVUSE_LOPP		date	False	
False	MOOTYHIK_KOOD		varchar(100)	False	Reference to measurement unit's classification element.
False	KL_VERSIOON_KOOD		varchar(100)	False	
False	LISAATRIBUUDID		xml	False	Additional attributes of classification in xml format, e.g.: '<Properties> <Colour>yellow<Colour> <Length>200</Length> </Properties>'
False	KLASSIFIKAATOR_KOOD		varchar(100)	True	Reference to classification where the element belongs to.

Table 16. Data structure of classification element

5.4.3 Table: KL_VERSIOON

Classification version.

Primary key	Attribute	Default value	Data type	Mandatory	Description
True	KOOD		varchar(100)	True	Code of classification version.
False	NIMI		varchar(200)	True	Name of classification version.
False	KIRJELDUS		varchar(4000)	False	Definition of classification version.
False	KEHTIV		int	True	Shows whether classification version is valid ('1') or invalid ('0').
False	KEHTIVUSE_ALGUS		date	True	
False	KEHTIVUSE_LOPP		date	False	
False	TYYP		varchar(50)	True	Shows whether classification version is 'time continuous' or 'versioned'. NB! Classification may simultaneously have both time continuous version and other version.
False	KLASSIFIKAATORI_KOOD		varchar(100)	True	Reference to classification.
False	YLEMINEKUTABEL		xml	False	Transition table or tables in xml format. Transition table shows how it is

					possible to convert classification elements from one version to another.

Table 17. Data structure of classification element

5.4.4 Table: SEOS

Relation between classification elements. There can be different relation types, but the system must allow adding them as well.

Primary key	Attribute	Default value	Data type	Mandatory	Description
False	KL_ELEMENT_KOOD_1		varchar(100)	True	Code of the first part of classification element.
False	KL_ELEMENT_KOOD_2		varchar(100)	True	Code of the second part of classification element.
False	SEOS_TYYP		varchar(100)	True	Relation type identifier.
False	KOMMENTAAR		varchar(4000)	False	Here the nature of relation is defined if necessary.

Table 18. Data structure of classification element

5.5 National requirements for defining metadata

On national level, data governance is organised by Statistics Estonia, who has compiled data governance management action plan (see [2]). Statistics Estonia is currently developing data definition standard. Metadata in agricultural big data system must also be in compliance with this standard. Statistics Estonia has prescribed DDI standard as a format for handling metadata (see [3]).

5.6 Requirements for service definitions

Creating public services of big data system requires existing service definition. The most important for every service is the definition of output metrics or what information this service produces. Similar list must be determined for input metrics. The cost of service depends to great extent on inputs. It is impossible to create a service if there is no place to retrieve input data for the service. This applies in particular to services that use bulk data. A simple calculator, where user enters input data when using the service, is not a tool for big data system. Service becomes the service of big data system when input requires using one or several data sources that allow retrieving bulk data automatically via web service or via regular file or database connection.

In order to give a clear overview of the services in big data system, system must define the content of services, which consists of the following:

Element	Description
Code	Unique service code across all services defined in the system.
Name	Name of service, e.g. “Humus balance calculator”.
Type	Type of service. Variants include: 1) “User service”. 2) “User interface” – visible and useable user access, i.e. service through which user can view and use data. 3) “Web service” – *web service* or infosystem-infosystem level machine interface, which is used to grant information system access to data source data.

Element	Description
	4) "Database" – database or open data file, set of data without specific user-oriented access service.
Definition	Brief detailed description of the service, including description of the reasons for creating the service, expected benefits and description of users.
Input and output	Definition of inputs and outputs. This must indicate definitions of input data and output data as definitions of metrics and dimensions.
Group	Service group code. It is reasonable to group services by theme. This facilitates systematisation of service definitions and finding duplicate definitions.
Owner	Service owner is an institution or a person who is involved in establishment and development of methodological content of the service.
Service administrator	Institution responsible for technical functioning of the service.
Establisher	Person or institution who proposed the idea of creating the service.
URL	Service link.

Table 19. Structure of service definition

Services are similarly defined in big data system to allow creating service catalog and its efficient search system.

5.7 Further recommendations for compiling definitions

Additionally, these aspects must be followed in case of definitions:

1) Definitions must not duplicate each other.
2) Definitions must be managed in uniform database.
3) When encoding services, it is not advisable to use only numeric codes, but instead add an abbreviation describing the object to make the code easier to read and reduce the number of errors when using codes.
4) All definitions must have OData web service, which allows using definition information in new services and data analyses.

5.8 Standards for web services

The following table describes the types of web services used in big data system.

Code	Name	Description	References
WMS	Web Map Service	Exchange of geoinformation in image format. The type of service is intended for downloading maps as images.	http://www.opengeospatial.org/standards/wms
WFS	Web Feature Service	Exchange of geoinformation in alphanumeric format, where information is transferred on the level of *feature* and *feature property*. This type of service is used for transferring geoinformation in detailed format, allowing consumer to decide how to display the information.	http://www.opengeospatial.org/standards/wfs
WCS	Web Coverage Service	Exchange of geoinformation in multidimensional format, where query dimensions can be pre-determined by the person making the query. Difference from other geoinformation services consists in its ability to send queries in parts.	http://www.opengeospatial.org/standards/wcs

Code	Name	Description	References
Point Cloud service	Point cloud service	Point cloud service is intended for exchanging spatial information as point cloud. For each point the service transfers X, Y, Z coordinates and any number of other parameters describing that point. This type of service is intended for use based on Lidar data and other data where measurement accuracy is related to a point and not a spatial area. Such structure is used e.g. in altitude model. The same structure should be used for soil composition and other datasets that support precision farming.	https://en.wikipedia.org/wiki/Point_cloud
RWS	RESTful web service	Type of web service where operations supported by it are standardised and information is transferred in text format. In particular, RESTful services can be used for transferring open data, where user session is not significant.	https://en.wikipedia.org/wiki/Representational_state_transfer
OData	OData RESTful web service	One widely supported REST service standard for sharing open data consists in Open Data web services. It was developed by OASIS and confirmed by ISO/IEC JTC 1. OData adapters are present in many data analysis software products such as Power BI and Tableau, which creates wide user community for services created based on OData standard. Moreover, it is rather easy to introduce OData service in any special software. The following aspects should be considered when planning the data structure of OData service: 1) Service must have not too deep or recursive data structure, in order to allow using the service with universal adapters in common analysis software. It must allow converting data into two-dimensional table, i.e. de-normalise without exponential increase in number of entries. 2) If de-normalising causes excessive number of entries (>5,000,000), the service must be divided in several parts, so that not all data objects are placed in one structure, but instead, it results in a partially de-normalised structure with relations between data of two different services. 3) Usability of the service must be tested on common analysis software products to detect critical data structure complexity, de-normalising of which is acceptable for the software.	https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=odata, https://www.oasis-open.org/news/pr/iso-iec-jtc-1-approves-oasis-odata-standard-for-open-data-exchange, https://www.odata.org/
X-Road service	X-Road service	Type of service used in Estonia, based on SOAP protocol that requires authentication and	https://www.ria.ee/et/riigi-

Code	Name	Description	References
		authorisation and corresponding infrastructure. In case of X-Road services, data structures are pre-defined and number of operations is unlimited. X-Road services are mostly used for moving confidential information.	infosusteem/andmevahetuskiht-x-tee.html

Table 20. Standards for web services

It has to be considered that many databases already have relevant services, and in that case, it is economically more viable to crate corresponding adapters instead of adding more services. This applies in particular to readymade software used e.g. in agrometeorology or agricultural machines.

5.9 Usage of data types

(Semantic) data types used on the level of the system definitions are described in the table below:

Data type	Explanation
Integer	Figure without decimal points.
Real number	Figure with decimal points.
Text	Text with random size. Text may also contain date or time.
Geometry	Bulk data describing the geometry of a phenomenon. More widely known as data type *geometry*. Geometry type is described in greater detail [6].

Table 21. Structure of service definition

Physical data models and databases use data types based on specific database engine.

5.10 Data quality standard

In order to ensure the quality of services in big data system, every interfaced service must pass data quality analysis.

Data quality analysis must be performed on the basis of real data and assessment must be provided to the aspects presented in the table below.

Code	Aspect assessed	Explanation
AKK-1	Compliance of RIHA data composition with real database	Does the data composition defined in RIHA comply with real database? Does the definition comply with the statutes of the database? Service of big data system cannot be established on illegally collected data. Legal background of data must be organised when creating the service.
AKK-2	Presence of data definition	Answer to the question concerning the real dataset of the database. Has the data model been documented?
AKK-3	Presence of data collection methodology and consistency	What methodology is used for collecting data? Is there methodological consistency?
AKK-4	Consistency of data format	Are data presented in consistent form? Do the data allow creating time series?
AKK-5	Reliability of data	Are the data reliable? Has the database taken steps to ensure data quality?

Code	Aspect assessed	Explanation
AKK-6	Conciseness of data	Are data presented in a concise manner, i.e. does data structure contain empty columns and tables and have the data in database presented in optimal volume?
AKK-7	Possibility of data transfer	Can data be transferred to another environment without any loss?
AKK-8	Machine processing of data	Do data allow machine processing?
AKK-9	Traceability of data	Are the data traceable in the database, i.e. is it possible to use them to understand processes represented by data from temporal and substantive aspect?
AKK-10	Integrity of data	Is the integrity of database data ensured on database level or by other methods?
AKK-11	Availability of data	Are data available? Do data transfer services have sufficient handling class?
AKK-12	Accuracy of data	Analysed according to opportunities and availability of information, including from other databases. Are data accurate, i.e. do they represent objects, subjects and events in real world accurately?
AKK-13	Comparability of data with data from other sources	Do the data of analysed source correspond to standards implemented in Estonia? Are personal identification codes and address data used in the same formats as they are used in other databases?
AKK-14	Usage of classifications and code lists	Which classifications are used for data, including: 1) Is there a solution for classifications management? 2) Is state classification used? 3) Is ADS used and to what extent (in percentage)? 4) Are the classification elements used in dataset also present in classifications? 5) Are classifications used in consistent manner? 6) Are used classifications time continuous or versioned?
AKK-15	Uniqueness of registration numbers	Does database use personal identification codes registration numbers, and are they accurate and unique?
AKK-16	Correct sequence of events related to object or subject	Are the events related to data object in logical and temporal order, e.g. are the dates when an animal was born and deceased in the right temporal order?
AKK-17	Completeness of data	What has been done on database level to ensure presence of mandatory data?
AKK-18	Accuracy of quantitative data when measuring quantity	Do quantitative parameters have positive value?
AKK-19	Presence of duplicates	Do data contain duplicates, i.e. does the register contain several equivalent entries concerning the same object or subject in real world?
AKK-20	Presence of entry statuses	Are entry statuses and states described accurately? Are the status or state data always present in an entry?

Table 22. Database data quality assessment criteria

The said aspects are partially based on RIA study [4].

5.11 Database quality assessment

Before interfacing database services with big data system, the database must undergo data quality assessment. The result of that assessment determines whether the database will be interfaced to big data system or not. If the quality corresponds to the purposes of using the data, it is necessary to establish data quality monitoring system. Monitoring system is based on the controls created during initial quality assessment, meaning that the same assessment will be carried out on a regular basis.

Main requirements for quality monitoring system:

1) System must support E scripts when defining control.
2) System must allow compiling data models subject to controls based on source data, taking into account relations between tables.
3) System must allow compiling and applying controls within one table and entry and also within several tables, by using columns of several tables simultaneously.
4) It must allow storing the control results in the system, as well as publish them in the same system and online by using common analytic visualisation elements (diagrams, graphs, tables).
5) System must allow displaying both original data and control results applied to such data.

6 Services of big data system

6.1 Choice of services

In the course of analysis, consortium members mapped over 60 potential services. There was a situation, where classification did not allow ascertaining the possibilities of socioeconomic use. At closer inspection it was discovered that it is reasonable to further group the services. After that thematic groups were determined, primary inputs necessary for creating the service were assessed and services ecosystem was described. This brought forward services without valid practice for compiling revenue estimate but valued by the consortium as vital components that ensure functioning of big data system as a whole.

The figure below describes whole context of agricultural big data, relations between the services to be created and relations that need to be considered when classifying the services.

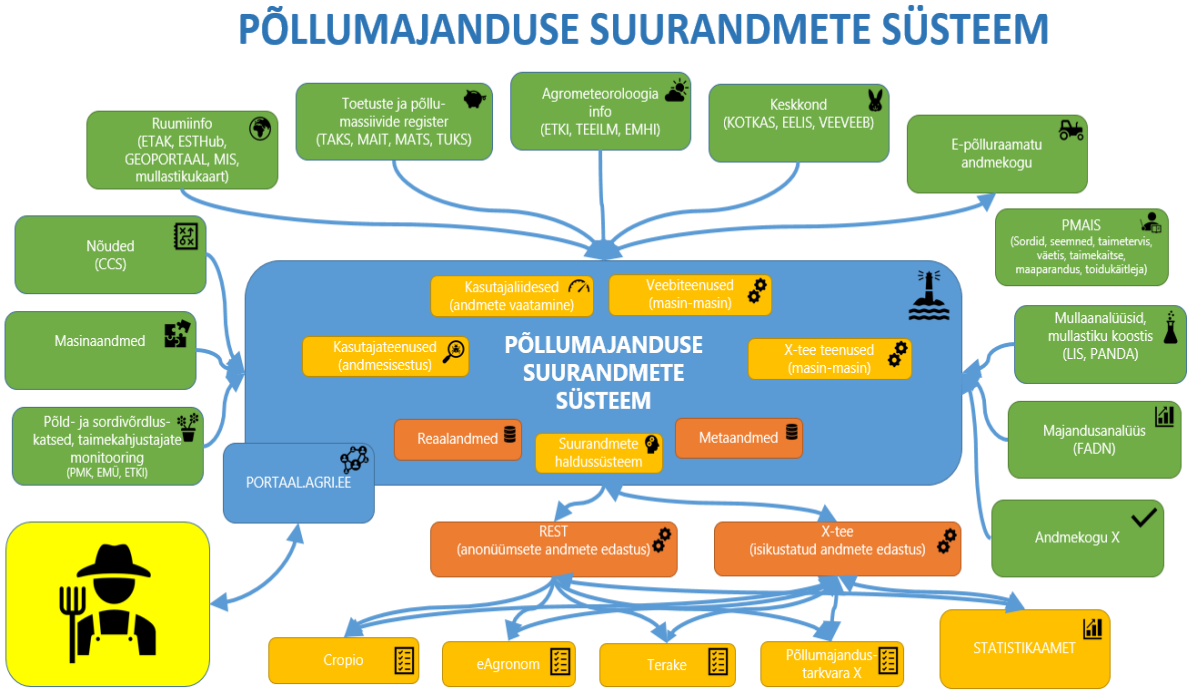

Figure 17. Agricultural big data system

Based on aforesaid, the services were grouped as follows:

- **Ecosystem of basic components** defines those components crucial for big data system that were most valued by the consortium. Ecosystem of basic components also considers technical services with the highest rating as prerequisites for new services to be able to enter the market. Primarily, the services that were basic components, but not technical services were chosen for economic analysis.
- **Production services** define significant services that belong to basic components but are substantive services with regard to agricultural production and have distinct economic revenue and cost component. Economic analysis was performed specifically for the services in this group.
- **Remote sensing services** define a group of remote sensing services, of which only one is in practical use, but several have background information provided by researchers that indicates great potential for extending the range of services. Unfortunately, it is difficult to assess its direct value by market players due to lack of practice.

- **Machine data services** define services related to automatization of inputs for many determined services that still have no solution on EU level, that are complicated due to legal regulations, but will create large comprehensive group of potential efficiency factors when realised.

6.2 Economic analysis of services

One of the goals of this present project "Long-term programme of knowledge transfer in the area of agricultural big data" is to analyse the benefits arising from the creation of agricultural big data system, what are the costs of creating the system and how feasible is the development of the system from economic aspect. Economic aspect is based on assessment of efficiency and effectiveness.

In order to determine the economic feasibility of big data system, an economic analysis was performed concerning the following aspects:

- What are big data system services and their interdependencies?
- What are the benefits or savings as a result of using the services?
- How much will it cost to create the services?
- What is the economic model of big data system like?

Data as information has primary importance from economic aspect, but in general, information cannot be used in agricultural sector without additional analysis and improvement of production activities to gain revenue. As the big data system is based on activities collected during primary production, which in turn comprise production technological activities, it is necessary to analyse the collected data, provide operative data-based services and thus, ensure input for further specification of production processes. Therefore, the basis for economic analysis consists in previously mapped services that would allow increasing the efficiency of plant production undertakings. Multidimensional economic analysis was performed for production services and certain remote sensing services (see choice of services). Analysis involved the following services:

- Plant protection advice and forecast; Plant pest mapping; Sharing information and providing forecasts of spreading of plant pests;
- Humus balance;
- NPK balance (integrated fertilising and liming suggestions);
- Electronic field data book (minimal functionality);
- Crop rotation planner;
- Integrated service of sharing agrometeorology data;
- Dissemination of information about requirements for aid applicants; Services for automatic submission of aid applications;
- Register services: Fertiliser register services; Web services of the register of plant protection products; Web services of plant variety register;
- FADN data visualisation service; Service for sharing FADN standard parameters; Calculator of economic size and production type (FADN)
- Variety selection and seed quality selection suggestions; Sharing data concerning variety comparison tests;
- Monitoring of waterlogged land (input for water indexes service);
- Yield forecast and quality model.

Economic analysis was carried out on the basis of assessing the significance of socioeconomic impact, with the inclusion of the following assessment components:

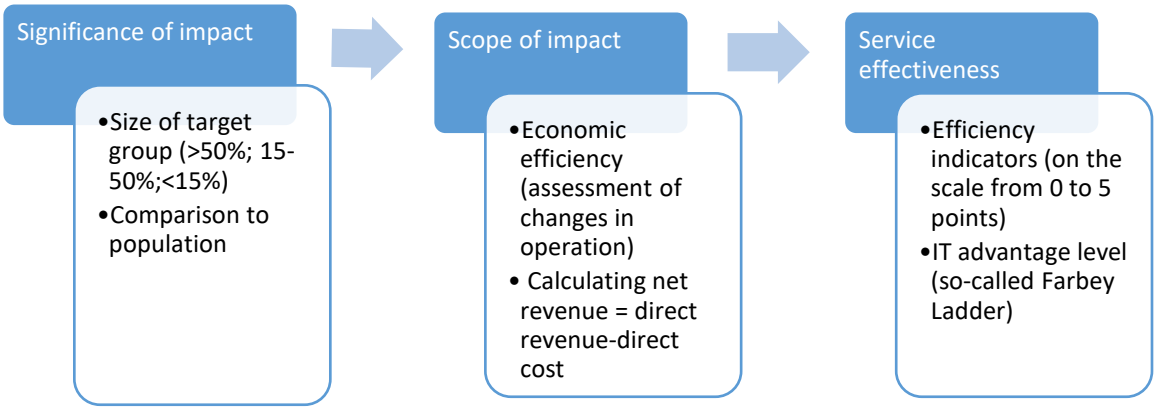

Figure 18 Components for assessing socioeconomic impact

In order to assess the aforesaid dimensions, it is necessary to know the services and users or target groups. The result of this present analysis was presented by using multidimensional table. For each service, assessment examined the parties and their relation to target group (producers). Parties included:

- Producer (target group);
- Consulting agency;
- Research institution;
- Financial institution;
- State agency;
- Input provider, third party.
- Service administrator

Significance of impacts depends on the size of target group and scope of impact. Target group is deemed big, if the share of persons using the service exceeds 50%. Average size of target group corresponds to a group with the number of service users ranging from 15 to 50% and group is considered small when the number of service users is less than 15%. Scope of impact was considered considerable if the operation of the target group (person, company, environment) may undergo significant changes compared to earlier, i.e. in case of significant benefit. Undesired impacts involve risk which was not considered in this analysis. Assessment of the scope of impact involved economic dimensions such as service efficiency and service effectiveness.

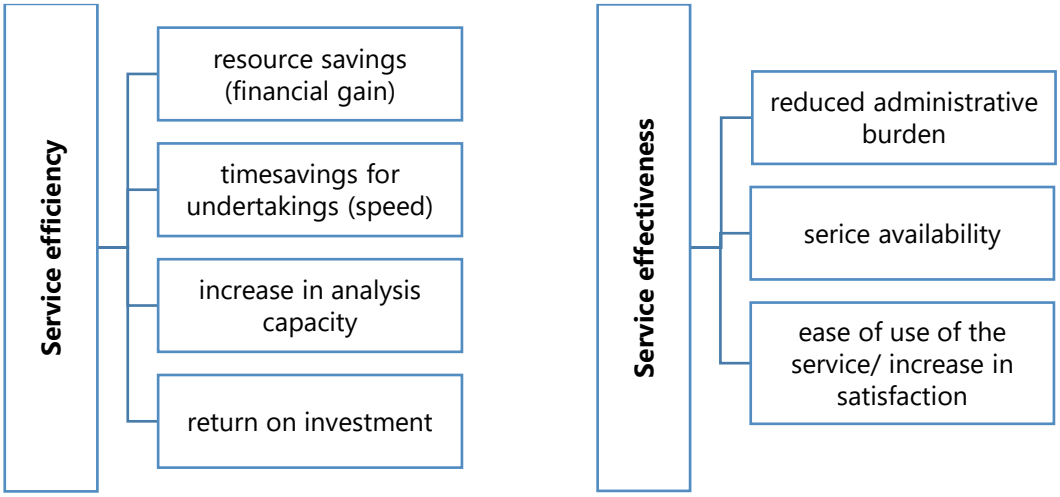

Figure 19. Possible dimensions and metrics for assessing economic efficiency and effectiveness

The metric used for assessment of service efficiency included both temporary and monetary benefits. Matching metrics included sales revenue (revenue from avoiding crop loss), savings in production inputs (benefit from savings on fertilisers and plant protection products) and revenue from preserving the environment. If there are other services added during the project, there might be additional metrics added as well.

In order to assess economic benefits, net revenue was calculated by using the following formulae:

Direct revenue = (quantity of estimated benefit × affected units (ha, producer) × cost) + (affected time unit (reduction of working time) × average hourly labour cost)

Direct cost = (service data acquisition, processing × hourly labour cost) + (service development × hourly labour cost) + (fixed cost of service × hourly labour cost)

Net revenue = direct revenue – direct cost.

Service effectiveness was assessed by using administrative burden and service quality and availability indicators. Additionally, the IT advantage ladder (so-called Farbey Ladder) discussed in PENG model was applied. The better the assessment to the system (in this case, to the service), the more complex and high-level opportunities the information system service provides. Effectiveness was assessed by groups of parties on the scale of 0 to 5 points and Farbey Ladder also by groups on the scale of 1 to 8 points, which in turn means that the score was summarised to get final result.

The following additional quantitative basic data were used for assessment:

- Number of producers in 2018; Crop area in 2018 (source: Statistics Estonia PM0281; PMS108; PRIA register of subsidies and fields 2018);
- Average yield in 2014-2018 (source: Statistics Estonia PM20);
- Average buying-in price in 2016-2019 Q (source: Estonian Institute of Economic Research 2019);
- Hourly labour cost by areas of activity in 2018 (source: Statistics Estonia PA001);
- Cost of input in schemes for calculating contribution margin in 2018 (Agricultural Research Centre 2018);

Results

Size of target group – Size of target group and thus, the extent of impact exceeds 50% in case of all mapped services and even ranged within 80-100%. Depending on parties, the size of target group (producers) still varies, according to conservative estimation, the interest of consulting agency may reach up to 50% and the interest of financial institution up to 60% from producers' population. Research institutions and state agencies are apparently interested in maximum share of producers' target group. According to socioeconomic analysis, target group is considered big if it exceeds 50%, hence the selected services have big scope. **Effectiveness** – In order to assess the scope of impact from social aspect, consortium members assessed the necessity and potential effectiveness of the service. Assessment was based on the definitions compiled when mapping services and possible positive changes. Additionally, the level of IT advantage impact (so-called *Farbey Ladder*) was determined. Services were assessed by related parties, resulting in summarised score. Based on effectiveness and level of IT advantage, the highest score was given to electronic field data book service, humus balance and NPK balance calculator with fertilising and liming suggestions and plant protection forecast and suggestions, but also sharing of data of plant variety comparison tests and plant variety selection suggestions. The latter received the same score as yield forecast and quality model.

Service	Assessment of target group size (3-big, 2- medium, 1-small)	Assessment of effectiveness (0-35 points)	Farbey Ladder (7-56 points)
Electronic field data book	big, up to 95%	**27**	**29**
Humus balance	big, up to 95%	**18**	**21**
NPK balance (can be integrated with fertilising and liming suggestions)	big, up to 95%	**18**	**21**
Plant protection suggestions and forecast; Mapping of plant pests; Sharing information about and providing forecast of spreading of plant pests	big, up to 95%	**17**	**21**
Sharing data of plant variety tests; Plant variety selection and seed quality selection suggestions	big, up to 95%	**17**	**21**
Yield forecast and quality model	big, up to 95%	**17**	**21**
Crop rotation planner	big, up to 95%	15	18
Service of sharing integrated agrometeorology data	big, up to 95%	15	14
Sharing information about requirements for aid applicants; Services of automatic submission of aid applications	big, up to 95%	14	14
Register services: Fertiliser register services; Web services of register of plant protection products; Web services of plant variety register (present)	big, up to 95%	20	14
FADN data visualisation service (+sharing standard parameters); Calculator of economic size and production type (FADN)	big, up to 100%	14	14
Monitoring of waterlogged land (input for water indexes service (included in set)	big, up to 95%	12	14

Table 23. Aggregated results of target scope and economic effectiveness

The next step was to carry out assessment of economic efficiency according to potential revenue and cost components of the service. Net revenue of services determined in the course of economic analysis is calculated for comparison purposes and thus, it must not be summarised by services. Services have mutual synergy and summarizing financial net revenue of all services requires additional calculations.

Efficiency – this factor was calculated by assessing direct revenue and direct costs and its output was net revenue. Development costs of services used imputed method, where development cost for a certain year was divided by useful life of potential service (on average 5 years), resulting in imputed development cost per one year. In conclusion, it is possible to achieve the greatest net revenue from the services of humus balance and suggestions concerning plant variety comparison tests and plant variety selection and seed quality selection (up to 24.6 and 23.0 million €/ year, respectively). These are followed by services of NPK balance and crop rotation planner. However, it is important to consider that the service of crop rotation planner requires the presence of electronic field data book or its equivalent. Electronic field data book has an important role from the viewpoint of target group and in terms of official data collection. As for potential revenue from electronic field data book, calculation of net revenue considered minimum functionality and thus, it does not reflect the revenue from services provided by private sector. Based on the distribution of revenue it is apparent that minimum functionality of electronic field data book primarily helps to save working time (up to 2.8 million €/year). Meanwhile, crop rotation planner is one of the possible services that relies on electronic field data book. Pursuant to initial calculations, implementation of electronic field data book with possible additional services may bring direct revenue in the sum of approximately 30 million €/a.

Service	Direct revenue, million €/year	Total cost of service, million €/year	Net revenue, million €/year	Net revenue range
Plant protection suggestions and forecast + Mapping of plant pests; Sharing information about and providing forecast of spreading of plant pests	up to 9.3	up to 0.08	up to 9.3	5,0-9,9 million
Humus balance	up to 23.1	up to 0.04	up to 23	>20,0 million
NPK balance (can be integrated with fertilising and liming suggestions)	up to 17.6	up to 0.04	up to 17.6	15,0-19,9 million
Electronic field data book	up to 2.8	up to 0.54	up to 2.3	1,0-4,9 million
Crop rotation planner	up to 15.1	up to 0.11	up to 14.99	10,0-14,9 million
Service of sharing integrated agrometeorology data	up to 0.7	up to 0.08	up to 0.6	<1,0 million
Sharing information about requirements for aid applicants; Services of automatic submission of aid applications	up to 2	up to 0.59	up to 1.4	1,0-4,9 million
Register services: Fertiliser register services; Web services of register of plant protection products; Web services of plant variety register (present)	up to 0.9	up to 0.2	up to 0.7	<1,0 million
FADN data visualisation service (+sharing standard parameters); Calculator of economic size and production type (FADN)	up to 0.3	up to 0.02	up to 0.3	<1,0 million
Sharing data of plant variety tests; Plant variety selection and seed quality selection suggestions	up to 24.9	up to 0.34	up to 24.6	>20,0 million

Yield forecast and quality model	up to 0.2	up to 0.08	up to 0.1	<1,0 million
Monitoring of waterlogged land (input for water indexes service (included in set)	up to 12.7	up to 0.59	up to 12.1	10,0-14,9 million

Table 24. Estimated direct revenue, cost and potential net revenue range attributed to services

For the purposes of complex socioeconomic assessment, net revenue ranges were attributed scores <1.0 million - 10p; 1.0-4.9 million -15p; 5.0-9.9 million - 20p; 10.0-14.9 million - 25p; 15.0-19.9 million-30p; >20.0 million - 35p. As seen from Figure 20, electronic field data book is one of the most significant services from socioeconomic aspect (71 points), despite the fact that minimal functionality does not allow achieving high net revenue.

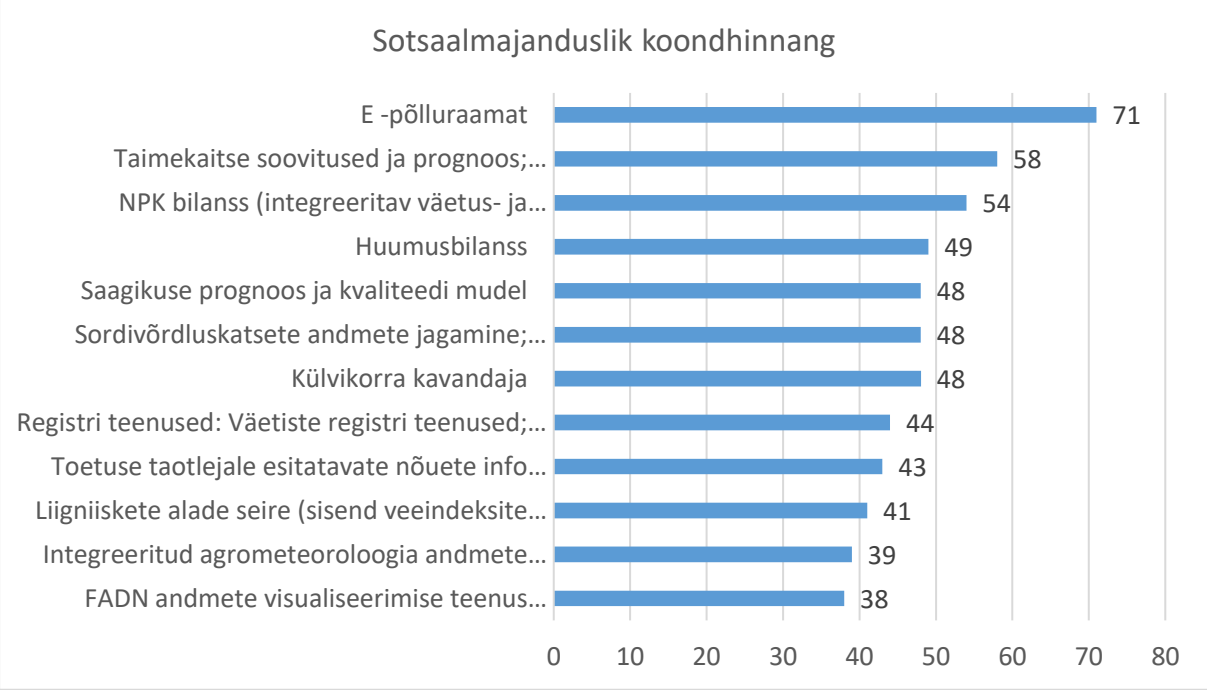

Figure 20. Consolidated socioeconomic assessment of selected production services

The following services allow avoiding crop loss, increase yield and minimise production risk. Revenue depending on time savings (Figure) is important primarily in case of services for sharing information on various registers, requirements and agrometeorological data. At the same time, this is a relative parameter, because total time spent has not been assessed in agricultural sector. Here, the basis consists in reduced time spent (e.g. for querying and reading requirements), exact total volume of which without IT service is not known. IT service helps to reduce time spent but does not eliminate it.

7 Roadmap for creating big data system

Task Name	Duration	Start	Finish
1. Big data system, Stage 0 / Preliminary activities (0-6 months)	**173 days**	**Mon 09.09.19**	**Fri 28.02.20**
Stage II of knowledge transfer project (preliminary activities)	**173 days**	**Mon 09.09.19**	**Fri 28.02.20**
Completion of conclusions of Stage I of knowledge transfer project	14 days	Mon 09.09.19	Sun 22.09.19
Completion of selections for organising procurement of Stage II of big data system	**14 days**	**Sun 22.09.19**	**Sat 05.10.19**
Agreement on architectural structure of big data system	7 days	Sun 22.09.19	Sat 28.09.19
Agreement on specified pre-selection of services to be realised	14 days	Sun 22.09.19	Sat 05.10.19
Agreement on services that can be provided by third sector	14 days?	Sun 22.09.19	Sat 05.10.19
Agreement on legal choices for Stage 0	14 days	Sun 22.09.19	Sat 05.10.19
Assessment of possibilities of alternative service managers (third sector)	**30 days**	**Sun 06.10.19**	**Mon 04.11.19**
Procurement of State II of big data system	**146 days**	**Sun 06.10.19**	**Fri 28.02.20**
Tendering authority compiles estimated scope of terms of reference for Stage II	30 days	Sun 06.10.19	Mon 04.11.19
Performance of procurement activities	88 days	Tue 05.11.19	Fri 31.01.20
Procurement is successful, procurement is not contested, and contract is prepared	28 days	Sat 01.02.20	Fri 28.02.20
Legal activities (Stage 0)	**160 days**	**Sun 22.09.19**	**Fri 28.02.20**
Ensuring legal support based on conclusions from Stage I and content in organising the procurement	160 days	Sun 22.09.19	Fri 28.02.20
Completion of the principles of intent for preparation of draft legislation	160 days	Sun 22.09.19	Fri 28.02.20
2. Big data system, Stage I / Main activities (6 months - 2 years)	**550 days**	**Sat 29.02.20**	**Tue 31.08.21**
Stage II of knowledge transfer project (implementation)	**550 days**	**Sat 29.02.20**	**Tue 31.08.21**
Procurement is declared successful and public contract is awarded	1 day	Sat 29.02.20	Sat 29.02.20
Organisation of big data system management	**549 days**	**Sun 01.03.20**	**Tue 31.08.21**
Establishment of organisation, roles, tasks	7 days	Sun 01.03.20	Sat 07.03.20
Involvement of controllers of existing databases	7 days	Sun 01.03.20	Sat 07.03.20
Management, development control, sustainability assurance	542 days	Sun 08.03.20	Tue 31.08.21
Ensuring that existing services are up to date and creation of new services	150 days	Sun 08.03.20	Tue 04.08.20
Big data basic system development according to the architecture selected in the terms of procurement	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
Creating environments	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
Creating data acquisition environment (depends on selected architectural solution)	180 days	Sun 01.03.20	Thu 27.08.20
Creating data processing environment (depends on selected architectural solution)	180 days	Sun 01.03.20	Thu 27.08.20
Creating publication environment (depends on selected architectural solution)	180 days	Sun 01.03.20	Thu 27.08.20
Compilation of data packages	180 days	Sun 01.03.20	Thu 27.08.20
Creating analytics and machine learning environment	180 days	Sun 01.03.20	Thu 27.08.20
Basic data management	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
Metadata management	180 days	Sun 01.03.20	Thu 27.08.20
Classifications and code lists management	180 days	Sun 01.03.20	Thu 27.08.20
Creating basic services	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
Creating database	180 days	Sun 01.03.20	Thu 27.08.20
Web services	180 days	Sun 01.03.20	Thu 27.08.20
User interfaces	180 days	Sun 01.03.20	Thu 27.08.20
User services	180 days	Sun 01.03.20	Thu 27.08.20
Realisation of the scope of services pursuant to the budget presented in procurement	**453 days**	**Sun 01.03.20**	**Thu 27.05.21**
Commencing service realisation based on the results of economic analysis and estimated volume fixed in public contract	**1 day**	**Sun 01.03.20**	**Sun 01.03.20**
Service 1 / Plant protection suggestions and forecast + Mapping of plant pests; Sharing information about and providing forecast of spreading of plant pests	1 day	Sun 01.03.20	Sun 01.03.20

Service 2 / Humus balance	1 day	Sun 01.03.20	Sun 01.03.20
Service 3 / NPK balance (can be integrated with fertilising and liming suggestions)	1 day	Sun 01.03.20	Sun 01.03.20
Service 4 / Service of sharing integrated agrometeorology data	1 day	Sun 01.03.20	Sun 01.03.20
Choosing database priority (Order of realising the scope)	**30 days**	**Sun 01.03.20**	**Mon 30.03.20**
Review of input relations by database or database group – database that has the most relations with services will be realised first	5 days	Sun 01.03.20	Thu 05.03.20
Agreement on the format of cooperation with database owners	7 days	Fri 06.03.20	Thu 12.03.20
Organising all accesses	30 days	Sun 01.03.20	Mon 30.03.20
Work related to databases	**439 days**	**Sun 15.03.20**	**Thu 27.05.21**
Database activities related to selected services	**137 days**	**Sun 15.03.20**	**Wed 29.07.20**
Eliminating the shortcomings indicated in the analysis of Stage I of knowledge transfer project	60 days	Sun 15.03.20	Wed 13.05.20
Realisation of interface needs of databases pursuant to standards	90 days	Fri 01.05.20	Wed 29.07.20
Integration of databases in the service (services 1-4)	**300 days**	**Sat 01.08.20**	**Thu 27.05.21**
T1: User interface: Plant protection suggestions and forecast + Mapping of plant pests	300 days	Sat 01.08.20	Thu 27.05.21
T1: User service: Plant protection suggestions and forecast + Mapping of plant pests	300 days	Sat 01.08.20	Thu 27.05.21
T1: Web service: Plant protection suggestions and forecast + Mapping of plant pests	300 days	Sat 01.08.20	Thu 27.05.21
T1: Database: Plant protection suggestions and forecast + Mapping of plant pests	300 days	Sat 01.08.20	Thu 27.05.21
T2-T4 interfaces (Humus balance, NPK balances, integrated agrometeorology)	300 days	Sat 01.08.20	Thu 27.05.21
Introduction of realised service	**122 days**	**Thu 01.04.21**	**Sat 31.07.21**
Information days for target group	21 days	Tue 01.06.21	Mon 21.06.21
Training for target group	40 days	Tue 22.06.21	Sat 31.07.21
Written materials and publications	122 days	Thu 01.04.21	Sat 31.07.21
Legal activities (Stage I)	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
Rules of procedure for the steering group have been established and members appointed	30 days	Sun 01.03.20	Mon 30.03.20
Establishment of general principles of big data system	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
General principles of activity of big data system, umbrella terms, lines of responsibility, data sharing rules and delegation of tasks to private sector	180 days	Sun 01.03.20	Thu 27.08.20
Analysis of opportunities to regulate big data system and establishment of legal choices for central technical solution of big data system and need for changes in legislation	**180 days**	**Sun 01.03.20**	**Thu 27.08.20**
Analysis of legal basis for processing data, description of the purpose of database management, decision on controllers and processors ad their tasks	180 days	Sun 01.03.20	Thu 27.08.20
Completion of analysis of delegating tasks to persons in private law	160 days	Sat 21.03.20	Thu 27.08.20
Establishment of data protection principles	180 days	Sun 01.03.20	Thu 27.08.20
Completion of description of data composition of big data system	180 days	Sun 01.03.20	Thu 27.08.20
Description of mandatory data producers and method for submitting information	180 days	Sun 01.03.20	Thu 27.08.20
Establishment of principles of issuing data and accessibility	180 days	Sun 01.03.20	Thu 27.08.20
Establishment of need for changes in legislation required to realise services	180 days	Sun 01.03.20	Thu 27.08.20
Preparation of draft legislation	**365 days**	**Wed 01.07.20**	**Wed 30.06.21**
Coordinating the draft legislation with ministries	**63 days**	**Wed 01.07.20**	**Tue 01.09.20**
Coordinating the draft legislation with the Ministry of Justice	30 days	**Tue 01.09.20**	Wed 30.09.20
Submitting the draft legislation to the Government of the Republic	1 day	Thu 15.10.20	Thu 15.10.20
Legislative proceeding of the draft legislation in Riigikogu	137 days	Fri 16.10.20	Mon 01.03.21
Development of implementing legislation	122 days	Mon 01.03.21	Wed 30.06.21
Big data system Stage II / Main activities (2 years - 5 years)	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Stage II of knowledge transfer project (follow-up activities)	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Potential joining of previously unrealised services with big data system	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Options for automatic joining (based on database analysis in Stage I)	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**

Structure and interface readiness of big data system is present	1 day	**Wed 01.09.21**	Wed 01.09.21
By following the joining standard established in Stage 0, joining databases required for providing new services to big data system with minimum time	1095 days	Wed 01.09.21	Fri 30.08.24
Activities targeted at third parties and creating third sector services	**180 days**	**Wed 01.09.21**	**Sun 27.02.22**
Introduction activities for potential new services target group organised by service owners	21 days	Wed 01.09.21	Tue 21.09.21
Finding new financing opportunities for creating new services	60 days	Wed 01.09.21	Sat 30.10.21
Realising services that were not realised in Stage II but are essentially necessary, according to budget	180 days	Wed 01.09.21	Sun 27.02.22
Ensuring that services are up to date, development activities	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Monitoring of existing services	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Feedback monitoring	1095 days	Wed 01.09.21	Fri 30.08.24
Data analysis and synthesis	1095 days	Wed 01.09.21	Fri 30.08.24
Proposals for amendments in services	1095 days	Wed 01.09.21	Fri 30.08.24
Monitoring of potential new services	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Feedback monitoring	1095 days	Wed 01.09.21	Fri 30.08.24
Data analysis and synthesis	1095 days	Wed 01.09.21	Fri 30.08.24
Mapping of new service opportunities	1095 days	Wed 01.09.21	Fri 30.08.24
Mapping of inputs for new research projects	1095 days	Wed 01.09.21	Fri 30.08.24
Communication with stakeholders in the context of potential services	1095 days	Wed 01.09.21	Fri 30.08.24
Service export opportunities	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Creating opportunities or introducing the services and preparations	1095 days	Wed 01.09.21	Fri 30.08.24
Service uniqueness monitoring within EU	1095 days	Wed 01.09.21	Fri 30.08.24
Legal action plan (Stage II)	**1095 days**	**Wed 01.09.21**	**Fri 30.08.24**
Legal analysis of the outcome of Stage I of big data system and development of changes in legislation if necessary	1095 days?	**Wed 01.09.21**	**Fri 30.08.24**
Stage II has been completed and big data system has been implemented	**1 day**	**Sat 31.08.24**	**Sat 31.08.24**

8 References

[1] Elcano: A Geospatial Big Data Processing System based on SparkSQL, Jonathan Engélinus and Thierry Badard Centre for Research in Geomatics (CRG), Laval University, Québec, Canada 2018 - https://www.scitepress.org/Papers/2018/67946/pdf/index.html

[2] Statistics Estonia. Action plan for Estonian data governance management - https://www.stat.ee/dokumendid/1413034

[3] DDI standard - https://www.ddialliance.org/

[4] RIA. Final report on data quality survey - https://www.ria.ee/sites/default/files/content-editors/publikatsioonid/andmekvaliteedi_uuringu_lopparuanne.pdf

[5] Open Data standard - https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=odata

[6] Description of geometric data type - https://www.postgresql.org/docs/9.4/datatype-geometric.html