The e-Science Central Study Data Platform

Paul Watson
School of Computing
Newcastle University
Newcastle-upon-Tyne, UK
paul.watson@ncl.ac.uk
and Alan Turing Institute, UK

Hugo Hiden
School of Computing
Newcastle University
Newcastle-upon-Tyne, UK
hugo.hiden@ncl.ac.uk

2022 IEEE 18th International Conference on e-Science (e-Science) | 978-1-6654-6124-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ESCIENCE55777.2022.00020

***Abstract*—This paper describes a novel study management platform that is being used to collect, process and analyse data gathered from a large-scale pan-European digital healthcare study. The platform consists of two main components. Firstly a secure, scalable, cloud-based platform to ingest and process data uploaded from body-worn sensors, as well as from clinical evaluation forms. Features extracted from this data are then loaded into a Data Warehouse with a novel schema designed specifically for study data. This allows scientists to explore, analyse and visualise this data in a variety of different ways. A key aspect of the warehouse design is that it also stores metadata describing the types and format of the data. This enables automatic report generation, exploratory data analysis and error checking. The overall result is a flexible, general purpose system that is open-source and uses the cloud for scalability. This paper describes the design of the integrated study data platform and its use in the large *Mobilise-D* study that has collected and analysed both sensor and clinical data from over 3,000 participants.**

***Index Terms*—data warehouse, healthcare**

I. Introduction

The system described in this paper is currently being used to manage data collected during Mobilise-D [8], a pan-European project to deliver a validated solution (consisting of sensor, algorithms, data analytics, outcomes) for real-world digital mobility assessment. It is doing this by validating digital outcomes that predict clinical outcomes for a range of important medical conditions: chronic obstructive pulmonary disease, Parkinson's disease, multiple sclerosis, proximal femoral fracture recovery and congestive heart failure.

In common with many large scale, distributed studies, the information to be managed and analysed is complex and highly heterogeneous. Specifically, it includes data that is:

- gathered from connected sensors worn by participants, such as accelerometers
- entered manually via forms submitted by clinicians during patient visits
- uploaded directly by external applications
- derived from source data by algorithms and analysis tools

The overall data flow is that once data has been ingested, it must be stored securely. Typically, there will then be some pre-processing (e.g., to extract features from time-series data generated by sensors). Finally, in order to analyse study data, scientists typically need to manipulate, visualise, analyse and explore this data in different ways, e.g. all results for one patient, all results for one type of measurement, or changes over time in one participant's measurements. There may also be a requirement to select and extract data to build models (e.g. through statistical methods or machine learning) that can then be used in healthcare applications that make predictions about a participant's future health.

Our experience is that each of these steps is often carried out in a laborious way that requires significant manual intervention. This slows down study progress, increases costs, and introduces the risk of human errors. For example, data collected in research studies is frequently stored in a collection of CSV files, often with metadata encoded in the filenames. This makes it difficult and time consuming for scientists to explore, interpret, share and analyse the data. For example:

"Files are grouped by folders with labels from 1-30 representing the participant number (30 participants in total). Each file was named systematically as '#-000_00B432**.txt', where '#' represents the walking surface condition and '**' represents the sensor location. For example, file '9-000_00B432CC.txt' stands for the trunk sensor ('CC') data while walking on the flat even surface ('9') for all participants." [4]

As a result, healthcare researchers spend a significant amount of time transforming and manipulating data in their analysis tools, such as *Excel*.

To overcome this, it is possible to use standard database design methods (e.g. entity-relationship modelling and normalisation [3]) to create a bespoke design for the structured data in a specific study. This has the limitation that each study will have a different schema that has to be designed, implemented and managed. Further, bespoke queries will be needed to extract and report on data. Similarly, bespoke checking and reporting code will also be needed.

Data Warehouses [7] have been successfully deployed in many application domains as a way to enable data to be manipulated and aggregated. Our first approach to this work was to explore whether we could use standard data warehouse design methods to meet our requirements for study data storage. We discovered that this was not possible as there is more heterogeneity in the types of healthcare data to be stored than there is in a standard data warehouse, in which there is only one type of *fact*. Even if databases or data warehouses were used to hold structured data, there still needs to be a

978-1-6654-6124-5/22/$31.00 ©2022 Crown
DOI 10.1109/eScience55777.2022.00020

solution for the vast amounts of raw sensor data now being collected in healthcare studies.

In this paper we describe our work to address these challenges through the design and development of an integrated study data management system; the *e-Science Central Study Data Platform* facilitates the automatic ingestion, processing, storage and analysis of study data, including structured data, sensor data and the features extracted from the sensor data.

The system consists of two main components. Firstly a secure, scaleable, cloud-based platform to ingest and process data streaming in from sensors, as well as from more static data such as clinical evaluation forms. Where required, processing is then applied to extract features from the data (e.g. from the time-series data generated by sensors). This data is then transformed into a structured form, and loaded into a Data Warehouse with a novel schema designed specifically for study data. This allows scientists to analyse and explore this data in a variety of ways. A key aspect of the warehouse design is that, as well as the data itself, the warehouse stores metadata describing the types and format of the data. This facilitates automatic report generation, exploratory data analysis and error checking. The overall result is a secure, flexible, general purpose system that is currently supporting a major pan-European study with over 3,000 participants, 100,000 items of data ingested (currently including 1TB of sensor data, but expected to grow to at least 6TB), and the execution of a set of sensor-data feature extraction algorithms written by over 50 developers.

This paper describes the motivation for, and the design of, this integrated study data system. The main contributions are:

- the design of the scalable, integrated study data management system.
- the design of a novel data warehouse schema for study data. A key feature is that the single-purpose fact columns found in a typical star schema are made generic by the use of measurement types and groups defined as required for a study. This ensures that the schema is flexible (it has been used to support a variety of studies)
- a demonstration of the capability of the system through its deployment in a large observational study project that is storing and analysing gait and clinical evaluation form data ingested from patients, clinicians and healthcare researchers from across Europe.

The rest of the paper is structured as follows. After reviewing related work in Section II, we describe the overall architecture of the *e-Science Central Study Data Platform* in Section III. This includes data ingest and scalable processing in the cloud. Section IV then introduces the design of the Data Warehouse component, before Section V explores its use of metadata to support automatic report generation, exploratory data analysis and error checking. We describe the system's use in the Mobilise-D Gait Analytics project in Section VI. This is followed by drawing conclusions and pointing to further work in Section VII.

II. Related Work

There has been other work on the design of study management systems. RedCap [13] is a secure web application for building and managing online surveys and databases. Unlike the e-Science Central Study Data Platform (e-SC) it is not open source, and there are strict licensing conditions that limit its use. It also lacks the support for time-series data offered by e-SC. Also, unlike RedCap, e-SC captures metadata in a way that enables generic tooling to be built (Section V-B).

i2b22 "Informatics for Integrating Biology and the Bedside" [19] is an open-source clinical data warehousing and analytics research platform that enables sharing, integration, standardisation, and analysis of heterogeneous data from healthcare and research. A star schema represents observations as facts, and has dimensions including patients and visits. The main difference from e-SC is that it lacks a way to group a related set of observations (e.g. all the observations collected in a clinical evaluation form). As will be described, we have found this to be invaluable for the analysis and reporting on study data. Unlike e-SC, i2b22 also lacks the ability to record the source of an observation (required for provenance) and to automatically check if data is within specified bounds.

The OMOP Common Data Model offers a standard way of representing data in observational databases. It can be used to federate data from diverse repositories by transforming data into the common data model, and by standardising on a common vocabulary for data represented within the model (e.g., terminologies, vocabularies, coding schemes). Open source tooling is available, including a library of standard analytic routines that exploit the common format. Unlike e-SC, OMOP separates observations and measurements, whereas our experience is that integrating them simplifies the warehouse schema, and analytics that often requires the two to be combined. It also lacks the ability to to group a related set of observations or measurements, and to check if data is within specified bounds.

All the above systems are focused on storing structured data in a database. This is not sufficient now that sensor data is a core part of many healthcare studies; for example Mobilise-D will gather at least 6TB of accelerometer data that needs to be stored and processed so as to extract features such as step counts and walking speed. For this reason, we found that no existing solution met our needs. We therefore designed the e-Science Central Study Data Platform to provide an integrated system that supports both the storage and processing of file based data (e.g. for raw sensor data), and a data warehouse for structured data (including features extracted from sensor data analysis).

e-SC is fully open-source, and flexible in terms of its deployment – to date we have deployed it on both Amazon [12] and Azure [11] clouds, with the warehouse in a PostgreSQL [10] database running in a virtual machine on Microsoft Azure, as well as on a managed PostgreSQL service running on the Amazon cloud. The data warehouse only uses standard RDBMS features, including standard SQL, and so can also be ported to a range of alternative database management systems.

III. The Design of the e-Science Central Study Data Platform

Figure 1 shows the cloud-based e-Science Central Study Data Platform (e-SC) that supports the ingestion, processing, analysis, querying and export of study data. Whilst a Data Warehouse is well suited to the querying and analysis of structured data, raw data (e.g. from sensors) must be collected, cleaned and processed before features extracted from it can be loaded into it. Data processing in e-SC is achieved by executing workflows which process data though a set of linked stages. For example, sensor data can be ingested and stored in e-SC, with its arrival triggering the automatic execution of a user-defined workflow that extracts features from it. For example, in the Mobilise-D case study presented in this paper, data from body-worn sensors (accelerometers) is loaded and a workflow used to calculate metrics such as stride length, walking speed and turns.

The system is scaleable as it can spread the execution of multiple workflows across a set of nodes in the cloud, which can be increased or decreased to accommodate changes in demand. The results from workflow execution are then stored as files in the e-SC data lake (Section VI).

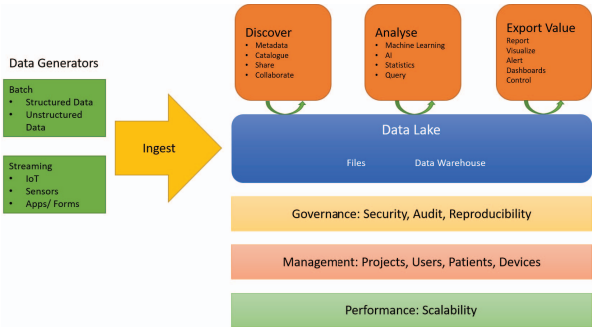

Fig. 1. e-Science Central Study Data Platform

A. Study Management

The Mobilise-D project supports a multi-cohort, multi-site observational study investigating the effect of a number of degenerative diseases on mobility and walking speed. The scale of this study means that there are a large number of distinct patient groups and associated collections of investigators and clinicians. Access to data and patients must therefore be carefully controlled, which necessitated the fine-grained security support offered by e-SC. Within the platform, disease cohorts and sites are separated into distinct e-SC studies. Within each study, users are divided into study administrators (who can manage all aspects of the study) and study members who can view any uploaded data. In addition, an *Uploader* role supports external data uploads by third parties via a published Application Programming Interface (API). This role provides no ability to view any of the uploaded information or retrieve data once it has been accepted by the platform.

Access to the platform is controlled via JSON Web Tokens which are managed using a web-based interface. Tokens can be issued to specific users with a defined validity period, and revoked to terminate access if that is required.

B. Data Acquisition and Validation

One of the strengths of the e-SC platform is its ability to manage heterogeneous data from a wide variety of sources and group it into records associated with a specific patient. To facilitate this, the platform draws a key distinction between structured observations and unstructured file-based data:

1) File Data: Unstructured file data is stored and versioned within the platform and physically persisted in cloud storage (e.g., Amazon S3). This data is arranged using a standard folder hierarchy, and provision is made for each patient within the platform to be assigned a private data folder for any uploaded data. In the case study presented in this paper, this feature is used to manage large accelerometer files that are uploaded via a public API.

2) Structured Data: Within e-SC, structured data observations are referred to as ”Events”. These Events are represented as JSON data [17]. Figure 2 shows the form of a simple Event capturing height, weight and BMI.

```
{
  "metadata": {
    "location": "Newcastle",
    "device": "Scales"
  },
  "data": {
    "weight": 83,
    "height": 179,
    "bmi": 28.8
  },
  "eventType": "Stats",
  "primaryKey": "height",
  "timestamp": 1570434690702
}
```

Fig. 2. Structured Event

The main contents of the Event are contained in the ‘data’ section of the JSON. Importantly, this data is validated by e-SC using a JSON schema when it is uploaded to the system.

The remainder of the Event data contains metadata and associated information, which allows e-SC to process and store the observation including:

metadata: A set of arbitrary name:value pairs that can be added to describe the Event. They are used to store additional data that may not be directly part of the validated observation, for example notes or information about any equipment that generated the Event.

timestamp: The UTC epoch timestamp capturing when the Event was collected.

eventType: A label defining the Event type. This provides a link to the JSON schema that is contained within e-SC and used to validate the Event as it is uploaded. This enables the contents and ranges for JSON documents to be specified and enforced. For example, the schema for an Event is shown in Figure 3; this can be used to validate a simple geographical co-ordinate. If the event type is not provided or the corresponding schema does not exist, the e-SC platform can be configured to reject the upload. Adopting a schema to validate data as early

as possible in the ingest process reduces the number of data errors that need to be found and corrected at a later date.

```
{
  "$id": "location.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Longitude and Latitude Values",
  "description": "A geographical coordinate.",
  "required": [ "latitude", "longitude" ],
  "type": "object",
  "properties": {
    "latitude": {
      "type": "number",
      "minimum": -90,
      "maximum": 90
    },
    "longitude": {
      "type": "number",
      "minimum": -180,
      "maximum": 180
    }
  }
}
```

Fig. 3. Example of a JSON Schema Document

To further reduce errors and development effort, e-SC makes use of a form generation library that automatically creates data entry forms from a JSON schema. This means that there is a single point in the system where data types and UI are specified. In the Mobilise-D project, these are used by clinicians to enter data on patients.

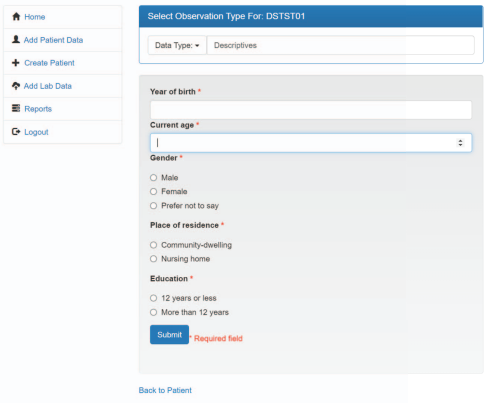

Fig. 4. Automatically generated form

Once data collected from a form has been uploaded and validated, it is stored as JSON data in a PostgreSQL database. This gives an opportunity, following a Study Change Request Process, to edit the data once it has been submitted to correct errors. This process is performed infrequently and is generally the result of user errors such as entering data that passes schema validation but is still incorrect, e.g., incorrect height and weight information or mistakes on free-text fields.

Once data is uploaded to the platform, it can be processed using the built-in data processing tools provided by the platform. The general approach to scalable Workflow processing in e-SC is described in detail in [1], while the workflows specific to the Mobilise-D project are described in Section VI.

C. Loading Data into the Data Warehouse

Periodically, code is executed within the e-SC workflow engine that iterates through all of the patients in all of the cohorts and loads their data into the Data Warehouse described in Section IV, where it is then used for study monitoring and the preparation of analysis reports for submission to the relevant medical regulators at the conclusion of the project.

The data warehouse is loaded from Events stored in the e-SC data lake using a loader helper library that extracts the appropriate fields from the JSON Events, formatts them correctly depending on their type and then inserts them into the warehouse (Section IV).

The process of loading data into the warehouse can either comprise a full drop/reload cycle that refreshes the entire warehouse, or an update process that only adds new or modified data. During the early stages of the Mobilise-D study where the Data Warehouse was being developed, and metadata describing the data to be collected in the study was evolving, the Data Warehouse was frequently regenerated from scratch. As the study progressed, smaller updates were required and it became preferable to apply new data in the form of incremental updates. A Web interface is provided so that study managers can directly initiate loading, reporting and querying operations (Figure 5).

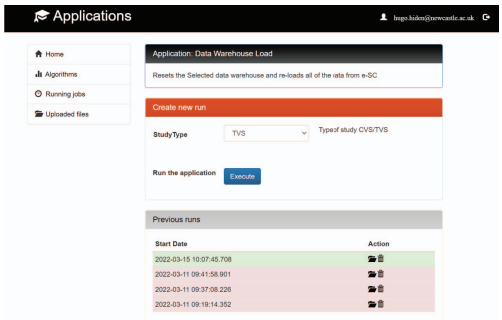

Fig. 5. Study Managers' Warehouse Load and Reporting Control Panel

IV. Data Warehouse Design

A key component of the e-Science Central Study Data Platform is the Data Warehouse. The design drew heavily on four guiding principles that were based on our experience of working on a range of digital healthcare projects:

- The data warehouse must be able to store any type of data collected in a study without modifying the warehouse schema. This means that when new types of data are collected in studies (e.g. from a new clinical evaluation form, a new data analysis program, or features extracted from data generated by a new type of sensor) they can be stored in the warehouse without any changes to its design. This has 3 main advantages: firstly, it enables us to fix and optimise the schema for the tables in which the data is stored; secondly it means that applications and tools (e.g. for analysis and error checking) built on the warehouse do not have to be updated when new types of data are added; thirdly, a single, multi-tenant database server can support many studies. This reduces the overall

costs, the start-up time for a new study, and the overheads of managing the warehouse.
- Descriptive metadata about the types of measurements must be stored in the warehouse so that tools (e.g. for error checking or reporting) or humans can interpret the data.
- The design should be optimised for query performance. In several cases, this has led to denormalization (duplication of data) to reduce the need for expensive joins.
- The warehouse must enable a security regime that restricts user access to the data.

Through this paper we use examples based on two common but distinct types of measurements that need to be stored in the Data Warehouse:

A. Clinical Evaluation Form Data

Healthcare information is often collected by clinicians or other healthcare workers who fill in a form while assessing a patient. The data is either directly entered electronically (e.g. on a tablet), or is initially written on paper before being transferred into an electronic system. Examples of this type of data are shown in Table I.

TABLE I
CLINICAL EVALUATION FORM DATA EXAMPLES

Identifier	Description	Type	Categories
C14.5	How often do you have to sleep sitting up in a chair because of shortness of breath?	Ordinal	Every night, 3-4 times a week, 1-2 times a week, less than once a week
G1	Participant has read PIS	Boolean	
G3	Year of Birth	DateTime	
GC1	Comorbidity	Boolean	
C5	Name of drug	Text	
C5.1	Dosage (mg)	Real	

B. Measurement Data

Increasing quantities and varieties of healthcare data are collected from sensors. As described in III, the raw sensor data is stored as files, analysed using workflows, with the derived clinical measurements being stored in the warehouse. Table II show an example of measurement data that has been derived from raw accelerometry data.

TABLE II
MEASUREMENT DATA EXAMPLE

Identifier	Description	Type
WB1	Avge Walking Speed	Real
WB2	Distance	Real
WB3	Stride Length	Real
WB4	Cadence	Real

The Schema for the Data Warehouse is shown in Figure 6.

Three key concepts embodied in the Warehouse schema design are now described.

C. Studies

All tables have a study identifier field; this is driven by the requirement to use a single data warehouse to store multiple studies. It enables a multi-tenancy approach in which the overheads of deploying and managing the warehouse can be shared across multiple studies. Organisations could chose to deploy only one warehouse for all their studies, or create one per study as they wish. The study id also provides the mechanism for managing security: a user's access can be restricted to a specific study or studies (for further security, the warehouse is implemented as an encrypted PostgresQL database). Each study consists of a set of measurements, and the metadata that describes them.

D. Measurements

Measurements are linked in related groups. For example, all the measurements collected from a form (e.g., as in Table I) or related features produced by analysing sensor data (e.g., as in Table II) would be contained in a single measurement group. This is a significant difference from the standard data warehouse design in which each entry in the measurement table is of one type, and independent – not part of a group of related measurements. It is this novel approach that gives the e-SC data warehouse its flexibility and generality.

Each individual measurement within a group is stored in the Measurement table, which holds the information shown in Table III.

TABLE III
MEASUREMENT TABLE FIELDS

Field	Description
id	a unique id
time	the date and time of the measurement
measurementgroup	the id of the group containing this measurement
groupinstance	a unique id for a set of measurements captured within a group. Each measurement within a group will be stored in a separate row of the measurement table, and so this field is used to link them together.
measurementtype	the id of an entry in the measurementtype table
participant	the id of an entry in the participant table
study	the id of an entry in the study table
trial	the id of an entry in the trial table; a study can consist of many trials
valtype	the id of the type of value
valinteger	the value for integer measurements, as well as booleans, nominals & ordinals
valreal	the value for real number measurements

To balance space efficiency with query performance, the measurement table only holds integers and real values: other, less common, types of values – *datetimes* and *text* - are stored in other tables with a link to the measurement table. This enables the selection and aggregation of numeric values – the most common types of values – without the need to perform expensive join operations. To allow users and tools to interpret nominal and ordinal data, entries are made in the *Category* table for each specific category. This table has the fields shown in Table IV.

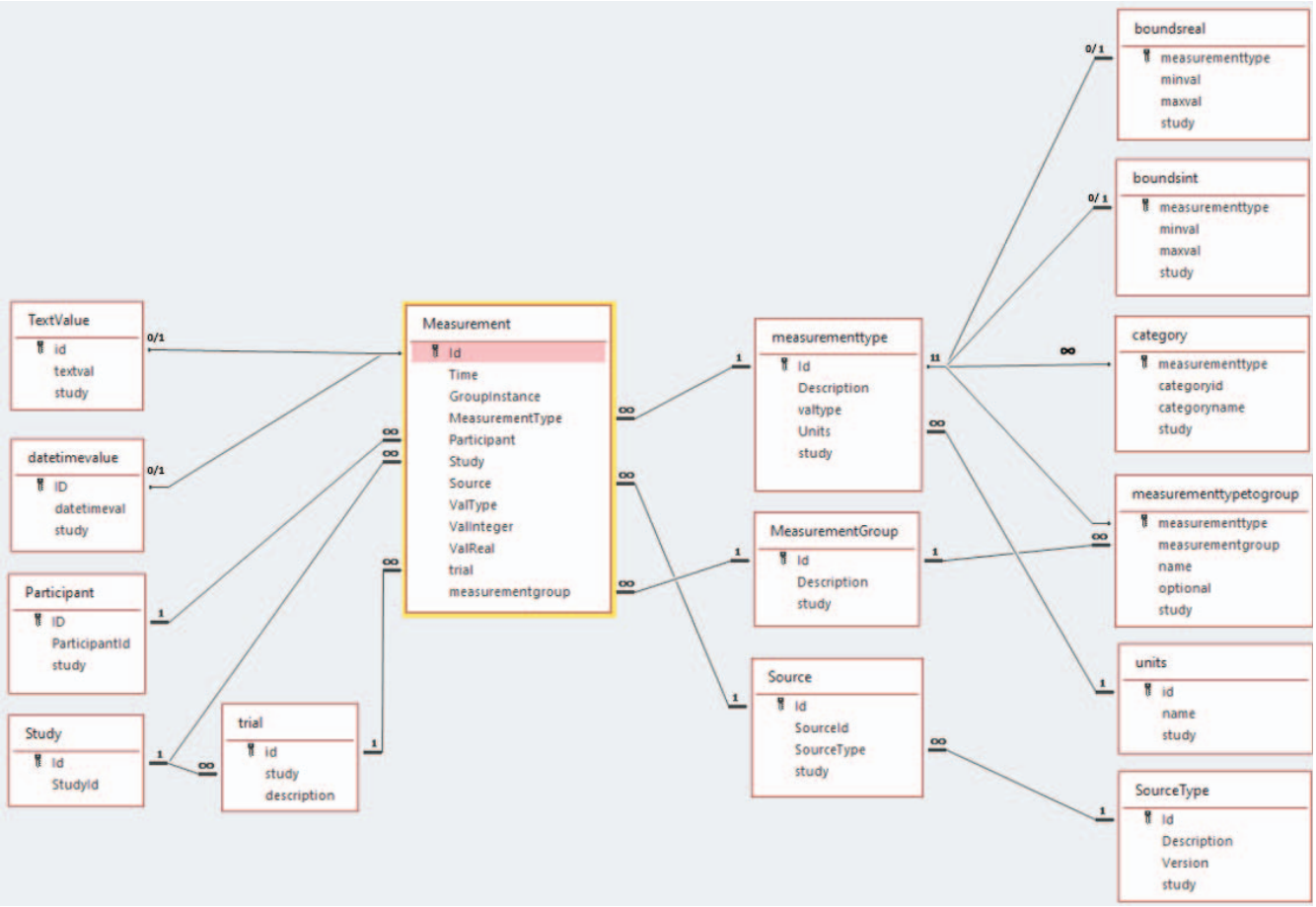

Fig. 6. Schema of the Data Warehouse.

TABLE IV
CATEGORY TABLE FIELDS

Field	Description
study	the study id
measurementtype	the id of the measurementtype
categoryid	a unique category id. For ordinal types, the order of the categoryid values is significant.
categoryname	a textual description of the category.

For example the categories of C14.5 – an ordinal value – from Table I would be stored as shown in Table V (where the measurementtype id is 8, and the study id is 1).

TABLE V
EXAMPLE CATEGORY TABLE ENTRIES

Study	Measurement Type	Category Id	Category Name
1	8	0	Every Night
1	8	1	3-4 times per week
1	8	2	1-2 time per week
1	8	4	Never in past 2 weeks

The participant table holds a unique entry for each participant in the study. For many studies, the optional *ParticipantId* field will hold a pseudo-anonymous id that the study managers can use to re-identify participants using a mapping table held in secure storage outside of the warehouse.

E. Metadata

A key design goal was that the warehouse must be able to store any type of data collected in a study without modifying the schema. However, it must also be possible for tools and humans to interpret the data. The data warehouse is therefore designed to enable the study manager to pre-configure the warehouse with metadata. The metadata describes the names and types of data that will be stored, and how data is formed into groups.

All the metadata is associated with a particular study so that multiple studies, each with different metadata, can be stored in the same multi-tenant data warehouse.

Configuring the metadata for a study begins with defining the type of measurements that will be stored. This is done by adding an entry in the *MeaurementType* table for each different

type of measurement found within the new measurement group. This table's fields are shown in Table VI.

TABLE VI
THE MEASUREMENTTYPE TABLE

Field	Description
id	unique id
study	study id
description	textual description of the measurement type
units	optional foreign key into the units table
valtype	the type of the value stored for these measurements

The *Units* table, referenced by the optional *units* field, enables the study manager to capture the specific type of units for the measurements that are stored. For example, the dose of a drug may be in mg. It is important to record this as without it there can be ambiguity in how to interpret a reading.

Once the basic measurement types for a study have been defined, related measurements can then be combined into measurement groups. Examples are all the measurements collected from a form (e.g., as in Table I) or related features produced by analysing sensors data (e.g., as in Table II).

A relevant *MeasurementGroup* id is stored with each measurement, enabling all the results from a specific group to be retrieved together from the warehouse and analysed. When a related group of measurements are stored, they are are also linked by the *GroupInstance*. This is unique to a specific instance of a group of measurements. For example, if multiple instances of the group of measurements shown in Table II are stored in the warehouse, they will all have the same *measurementgroup id*, but each instance will have a different *GroupInstance* value. This enables all the measurements held in the same group instance to be retrieved together.

To improve query performance through denormalisation, all metadata is associated with a single study. However, tooling is provided to make it easy for study managers to set up multiple studies with the same metadata.

F. Provenance

The data warehouse was designed to support reproducible science through capturing provenance information. Each measurement can therefore have *Source* information associated with it consisting of a unique source id. For measurements collected through a clinical evaluation form, this could be a unique id for the form, while for features extracted from data collected by a sensor, this could be the sensor id. This information is held in two tables: *Source* and *SourceType*.

The *SourceType* Table holds generic information about a type of source. This includes a textual description and a version (e.g. McRoberts MoveMonitor, v1.2).

The *Source* table's fields include a link to the relevant entry in the *SourceType* table as well as a unique identifier for the source (e.g., a device number).

Each entry in the *Measurement* table can link to the relevant entry in the *Source* table. This creates a provenance trail to a specific sensor, or a specific version of an algorithm that generated a measurement. As well as supporting reproducibility, this can also be of benefit when an bug is found in a version of an algorithm, or a fault found in a specific sensor. Because of the information held in the *Source* table, it is then straightforward to identify all measurements that were generated by the failing hardware or software.

G. Data Warehouse Python Client Library

A Python client library is provided which can retrieve, filter, and analyse data from the warehouse without the need to write SQL or understand the schema of the data warehouse. The code is open source, and made available as the data-warehouse-client package that can be installed through pip from PyPi [14]. This library provides access to the most frequently executed queries and can be used to search for patient specific data, all data of a certain data type, produce summary reports of data set completeness and export selected fields in comma separated variable format for import into external analysis packages. The provision of such a library opens up the entire set of python machine learning, visualisation and reporting tools and gives a relatively straightforward way to create sophisticated, bespoke study monitoring reports.

V. EXPLOITING METADATA IN THE DATA WAREHOUSE

A key feature of the e-SC data warehouse design is that it holds metadata as well as the data itself. This enables humans and automated tools to interpret and process the data. In this section we give three examples of the automated tooling that the data warehouse offers through exploiting metadata.

A. Automating Data Veracity Checking

A major challenge for any store of data collected from experiments is to ensure the veracity of the data. In this section we describe how the design of the e-SC Data Warehouse enables it to automatically identify a range of types of errors in the data.

The first line of defence against incorrect data being stored in the warehouse comes from placing constraints on foreign keys – the links between tables shown in Figure 6. For example, it is impossible to insert a new measurement into the warehouse if its *MeasurementType* field does not correspond to an entry in the *MeasurementType* table, nor if the *MeasurementGroup* field does not correspond to an entry in the *MeasurementGroup* table, nor if the *Participant* field does not correspond to an entry in the *Participant* table, nor if the *Study* field does not correspond to an entry in the *Study* table.

The metadata that is defined by the study manager then offers further ways to automatically check veracity. Errors are detected through checks for the following:

1) measurement types declared as *Ordinal* or *Nominal* but without corresponding entries in the *Category* Table
2) measurements declared as *Ordinal* or *Nominal* but that that refer to a non-existent category
3) measurement declared as being of type bounded integer, but without entries in the *BoundsInt* table which table (which holds the maximum and minimum values that the integer can take)

4) measurements declared as *Bounded Integers* whose value is outside of these bounds
5) measurement types declared as bounded real but without entries in the *BoundsReal* table (which holds the maximum and minimum values the real can take)
6) measurements declared as *Bounded Reals* whose value is outside of these bounds
7) measurements where the *Value Type* does not match the value stored in the *Measurement* table
8) non-optional measurements missing from a measurement group instance. Measurements can be declared as optional in the *MeasurementTypeToGroup* table; this means that they do not have to be included in a measurement group instance. However, if they are not declared as optional, but are missing from a measurement group instance, then this check will detect the error.

Checks 1, 3 and 5 can be made once the metadata for a study has been configured, while 2, 4, 6, 7 and 8 can be made either when measurements are inserted, or (as in the current implementation) once each batch of measurements has been ingested. These checks are all performed automatically, with no configuration or extra information needed by the study manager. This is a significant advantage that comes from storing a rich set of metadata in the data warehouse.

B. Automated Study Reporting

Because the metadata and data for all studies are represented in a standard way, it is possible to write generic report generators. These take a list of measurement groups or measurement types, and produce either a printed report or a *CSV* file that can be loaded into another tool. An example report used in *Mobilise-D* project is shown in Figure 7. This shows the values of some of the types of measurements stored in the data warehouse for each participant in a study (represented by their unique, pseudo-anonymised id). In this report, the *Cohort* represents the participant's condition (e.g., PA is Parkinson's Disease), while the *Site* indicates the clinical research group who are collecting the measurements. The other columns show data that is relevant to the analysis of the participant's gait.

Researchers running a healthcare study involving human participants also need to track the progress of signing-up participants, and collecting data from them. The Data Warehouse can therefore also generate a range of reports on this, including those identifying missing and duplicated data for each participant.

C. Automating Exploratory Data Analysis

Tooling has been written that exploits the Pandas Profiling Library [20] to automatically generate an exploratory data analysis report for each measurement group. Part of an example is shown in Figure 8. This enables scientists to quickly view a range of useful features including missing values, the distribution of each measurement type, and the correlations between different measurements. This is used by both study managers and scientists to identify both issues and interesting features in the data.

Id	Cohort	Site	Age	Gender	MOCA	Falls	Walking Aids	General Pain	FVC	Distance
1	HA	SA	65	F	29	0	N	33		
2	HA	SA	67	M	29	0	N	1		
3	HA	SA	65	F	28	0	N	10		
4	PD	SA	52	M	28	6	N	12		
5	PD	SA	57	F	29	0	N	19		
6	PD	SA	69	M	25	0	N	1		
7	COPD	SA	79	M	20	0	N	6	2.81	348.5
8	COPD	SA	75	F	29	0	N	18	3.8	380
9	HA	SA	65	F	28	0	N	8		
10	PD	SA	71	F	20	0	N	3		
11	HA	SA	71	M	30	0	N	2		
12	HA	SA	73	F	29	0	N	12		
13	COPD	SA	63	M	26	1	N	50	3.08	380
14	PD	SA	66	F	30	0	N	39		

Fig. 7. First Rows of an Example Report

Fig. 8. Example Profile for Exploratory Data Analysis

VI. Clinical Application: Mobilise-D Gait Analytics

One of the 21st century's greatest challenges is finding the optimal treatment for people with impaired mobility resulting from ageing and chronic disease. To address this challenge, the *Mobilise-D* [8] consortium has been formed. It consists of 35 partners from 13 countries. Cohorts of participants at 18 sites are assessed through a combination of clinical evaluation forms (collecting information such as that in Table I) and measures extracted from experiments in which participants wear sensors for extended periods – Table II showed an example of these measures. In this section we describe how the e-Science Central Study Platform underpins the storage, sharing and analysis of data in the *Mobilise-D* project.

As shown in Figure 9, data is uploaded to e-SC via either an API (e.g., sensor data) or a web application (e.g., clinical evaluation forms) and stored in the e-SC *Data Lake*. The data is associated with a specific patient through the use of a pseudo-anonymous patient identifier generated by the study co-coordinators. There are three basic types of data that are

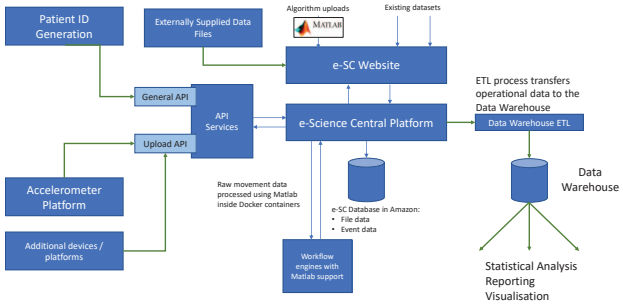

Fig. 9. Data Ingest Routes

ingested:

- **Clinical Observations**: Clinicians complete pre-defined forms during patient visits using a custom application running on tablets. These forms are represented as JSON structured data that are validated using JSON schema during upload to e-SC (as described in Section III.
- **Derived Results**: Raw accelerometer data files, collected during a 7-day period, are ingested and processed using e-SC workflows as described below.
- **Additional Data**: The results derived from the accelerometer data can be highly dependent on external factors such as weather. For example, during poor conditions, there is a likelihood of observing a decrease in mobility that may not be associated with the underlying medical condition. Local environmental data is therefore collected via a mobile phone along with GPS data, and uploaded to the platform via an API.

Currently there are over 100,000 measurements in the warehouse (including 1TB of sensor data), from 26 different measurement groups. Data has currently been collected from over 3,000 participants. The data is then loaded into the data warehouse, where a range of validation checks are run (described in Section V). Reports are also automatically generated for those running the studies to inform them of the progress in collecting data, while other reports are generated for scientists analysing the data (Section V).

A. Calculation of Derived Results

The focus of the Mobilise-D project is to derive clinical measures that use accelerometry data to describe the progress of the various conditions. There are a number of methodologies that have been used to do this in the relevant literature and these consist of several calculation steps that must be performed, each of which have several candidate algorithms and implementations. A key component of the project has been to systematically benchmark the various approaches and identify the most accurate approach for each medical condition. This requires large parameter sweeps to be performed over the collected data using a large number of algorithm permutations. This would have been an impossible task to perform on a single workstation as each 7-day set of accelerometer data takes several hours to process. Therefore, an approach that could perform these calculations in parallel was required. The e-SC workflow engine supports the required scaling – it has been shown to be capable of analysing large sets of sensor data in previous projects [15].

Workflows were created, making use of software packaged to run within Docker containers, and executed directly using the e-SC workflow engine. These containers are subject to the standard e-SC security model and so only authenticated e-SC users are allowed to upload containers and use them in workflows. This approach meant that accelerometer data processing pipelines comprising code written in multiple languages (typically a combination of Matlab, Python and Java) could be created and executed. This benchmarking exercise packaged code from over 40 individual contributors into more than 50 distinct processing blocks. Whilst a full evaluation of the various algorithms is beyond the scope of this paper, a comprehensive review has been performed [21].

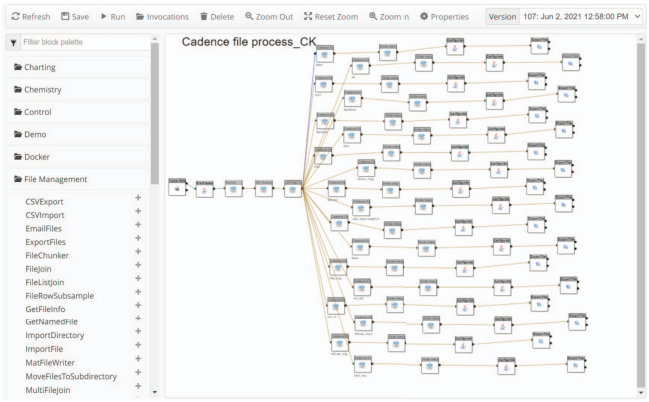

Fig. 10. Parameter Sweep Workflow

Figure 10 shows a workflow that processes a single accelerometer file using a set of different algorithms. This workflow was typical of those used during algorithm benchmarking and the e-SC platform allowed a workflow written to operate on a single file to be executed automatically for all of the accelerometer files for each patient using multiple cloud workflow engines. This approach has been demonstrated for runs using several hundred servers in [2].

The workflows output a summary of key walking characteristics such as Gait speed and number of walking bouts. This data was formatted into JSON Events with associated schema such that they were suitable for ingestion directly into the Data Warehouse. This process resulted in a comprehensive set of study data in the Data Warehouse, including both clinical observations and the derived Gait data. The reports required by the scientists for regulatory submission can then be generated.

VII. Conclusions

A key requirement for large healthcare studies is a computing infrastructure that can store, share and analyse a range of data collected from participants or other sources, often nowadays including sensor data. We have found that this is usually done using a range of non-integrated technologies that place a large burden on manual processes that are both time-consuming and error prone. In this paper, we have described an alternative – the e-Science Central Study Data Platform. This offers an integrated solution that is currently supporting the *Mobilise-D* project – a large, multi-site pan-European study in which large quantities of data are being collected from over three thousand participants from over ten countries. The study has generated over 100,000 measurements from 26 different measurement groups, including data derived by using the platform to analyse 1TB of sensor data using algorithms written by over 40 individual contributors.

A key component of the system is a data warehouse with a novel schema that enables heterogeneous healthcare data to be stored, queried and analysed. An important design feature is that metadata is held in the warehouse, in addition to the data itself. This has proved extremely valuable for many purposes, including enabling users to interpret data, automatically checking the veracity of data, checking for missing data in studies, automatically tracking the progress of studies, and automatically generating exploratory data analysis reports. The Python client library developed to interact with the warehouse has also provided an effective way for programmers to access data without them needing to understand the schema or write SQL in the vast majority of cases.

When a study is collecting sensor data – as is the case with *Mobilise-D* – the scalable workflow engines of the e-Science Central Study Data Platform have proved invaluable for deriving features from vast quantities of time-series data (e.g. step counts, turns, walking speed). These features are then inserted into the data warehouse for analysis by scientists. As the platform runs in the cloud, the number of nodes processing data can be scaled up to ensure that data is processed in a timely fashion [1].

In order to test the generality of the the e-Science Central Study Data Platform design we have also deployed it in other projects, including for anonymised public transport passenger data collected during the COVID-19 pandemic [9]. Data on passenger numbers and locations in trains and busses was analysed to give a measure of social distancing.

The e-Science Central Study Data Platform is open source, and so can be self-hosted, but we also provide it as a cloud-based service to support other projects. Software as a Service has been a major trend in computing over the past 10 years and providing a study data platform in this way reduces the cost, effort and skills required by those working on healthcare and other projects that need to store, share and analyse data.

After experimentation across a range of projects, the data warehouse schema design is now stable, and so we are focusing on user tools that exploit its features. This includes those for visualization, and for deploying statistical methods and machine learning to build models over study data.

Acknowledgment

We thank our funders. This work is partly supported by: Mobilise-D. A project funded by the Innovative Medicines Initiative 2 Joint Undertaking under grant agreement No 820820. The Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA.

TRACK (Transport Risk Assessment for COVID Knowledge). A project that is funded by UKRI (UK Research and Innovation) grant EP/V032658/1.

References

[1] Hiden, H., Woodman, S., Watson, P. and Cala J., "Developing cloud applications using the e-Science Central platform," Philosophical Transactions of the Royal Society A, Volume 371, Issue 1983, 2013.

[2] Cała, J., Hiden, H., Woodman, S., Watson, P., "Cloud computing for fast prediction of chemical activity, " Future Generation Computer Systems, 29(7), North Holland, 2013 Future Generation Computer Systems 29 (7), 1860-1869

[3] Codd, E. F. (1990). "The Relational Model for Database Management (Version 2 ed.)," Addison Wesley Publishing Company. ISBN 978-0-201-14192-4.

[4] Luo, Y., Coppola, S.M., Dixon, P.C. et al. A database of human gait performance on irregular and uneven surfaces collected by wearable sensors. Sci Data 7, 219 (2020)

[5] Thakar A., Szalay A., Kunszt P. and Gray J., "Migrating a multiterabyte archive from object to relational databases", Computing in Science and Engineering, vol 5(2), 2003.

[6] Stonebraker, M., Becla, J., DeWitt, D. and Lim, K-T., Maier, D., Ratzesberger, O. and Zdonik, S., "Requirements for Science Data Bases and SciDB", CIDR 2009 - 4th Biennal Conference on Innovative Data Systems Research, 2009.

[7] Kimball R., Ross, M., "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition," Wiley, 2013.

[8] Rochester L, Mazzà C, Mueller A, Caulfield B, McCarthy M, Becker C, Miller R, Piraino P, Viceconti M, Dartee W, P, Garcia-Aymerich J, Aydemir A, A, Vereijken B, Arnera V, Ammour N, Jackson M, Hache T, Roubenoff R., "A Roadmap to Inform Development, Validation and Approval of Digital Mobility Outcomes: The Mobilise-D Approach," Digit Biomark 2020;4(supplement 1), pp 13-27.

[9] Noakes C., "TRACK: Transport Risk Assessment for COVID Knowledge," UK Research And Innovation Project EP/V032658/1, https://gtr.ukri.org/projects?ref=EP%2FV032658%2F1

[10] The PostgreSQL Global Development Group, "PostgreSQL 13.2 Documentation,", 2021

[11] Microsoft, "Azure Developer Guide," https://docs.microsoft.com/en-gb/azure/guides/developer/azure-developer-guide, 2021

[12] Amazon Web Services, "Amazon RDS for PostgreSQL," https://aws.amazon.com/rds/postgresql/, 2021

[13] Redcap, "Research Electronic Data Capture". RedCap project, https://www.project-redcap.org/

[14] Data Warehouse Client Python package, data-warehouse-client, https://pypi.org/project/data-warehouse-client/

[15] Zhu G., Catt M., Cassidy S., Birch-Machin M., Trenell M., Hiden H.G., Woodman S.J., Anderson K.N., "Objective sleep assessment in¿ 80,000 UK mid-life adults: Associations with sociodemographic characteristics, physical activity and caffeine", PLoS One, 2019.

[16] University of Wisconsin–Madison, "HTCondor", https://htcondor.org/.

[17] "Javascript Object Notation", https://www.json.org.

[18] "Json Schema Specification", https://json-schema.org/specification.html.

[19] "i2b2 Community Wiki", https://community.i2b2.org

[20] "Pandas Profiling Library", https://pypi.org/project/pandas-profiling/

[21] Mazzà C, Alcock L, Aminian K, et al, "Technical validation of real-world monitoring of gait: a multicentric observational study", BMJ Open 2021;11:e050785. doi: 10.1136/bmjopen-2021-050785