Advanced Data Management 2017/2018

Team: Andrea Canepa, Stefano Rebora

PART I

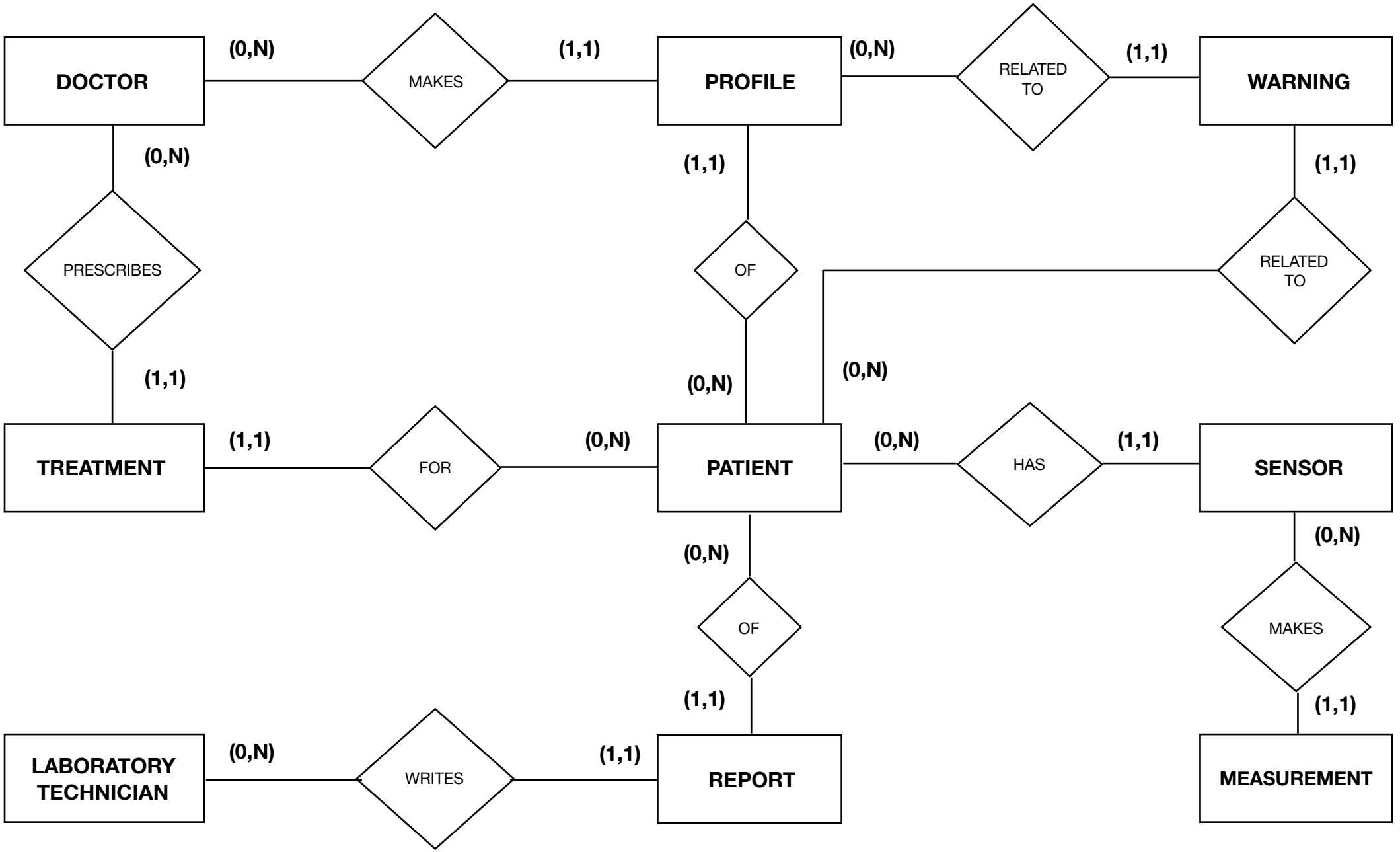
OVERVIEW

The reference domain concerns a remote patient monitoring service. In this context, we are interested in two main applications: a mobile application for the patient and a desktop application for the hospital/clinic staff.
Each patient is equipped with one or more medical sensors measuring physiological parameters to perform periodically routine tests and the data are sent to the hospital/clinic database by the mobile application. Data are stored and are not managed as a stream.
The hospital/clinic staff can visualize and analyze the data, write periodically reports that summarize the general patient state of health and suggest a specific treatment.

Typical workloads mix together write and read operations.
The aim of this project is the design and the implementation of the distributed data management layer for the patient mobile application and the staff desktop application. To this aim, we choose Cassandra as reference technology and we rely on CQL for workload implementation.

APPLICATION REQUIREMENTS

The mobile and desktop application need a database containing data about doctors, laboratory technicians, treatments, reports, patients, patient profiles, warnings, sensors and their measurements.
Here you can find a sketch of the ER diagram modeling the entities:


DOCTOR
(0,N)
MAKES
(1,1)
PROFILE
(0,N)
RELATED TO
(1,1)
WARNING
(0,N)
PRESCRIBES
(1,1)
(1,1)
OF
(0,N)
RELATED TO
(1,1)
(0,N)
TREATMENT
(1,1)
FOR
(0,N)
PATIENT
(0,N)
HAS
(1,1)
SENSOR
(0,N)
(0,N)
OF
MAKES
(1,1)
(1,1)
LABORATORY TECHNICIAN
(0,N)
WRITES
(1,1)
REPORT
MEASUREMENT

Doctors

A database administrator creates an account for each doctor of the hospital through the desktop application. Each account has the following information:

1. *doctor_ID*
2. *password*
3. *name*
4. *surname*
5. *date_of_birth (represented by an integer value following this schema: YYYYMMDD. For example "19951020" means 20 September 1995)*
6. *specialization (e.g. "Cardiologist", or "Orthopedic", ..)*

The attribute doctor_ID has to be unique.

Doctors can change their password through the desktop application, providing their current password and the new one. The application must only update the password if the correct current password is provided.

Laboratory Technicians

A database administrator creates an account for each laboratory technicians of the hospital through the desktop application. Each account has the following information:

1. *technician_ID*
2. *password*
3. *name*
4. *surname*
5. *date_of_birth (represented by an integer value following this schema: YYYYMMDD. For example "19951020" means 20 September 1995)*
6. *laboratory_code (e.g. 'BIOLAB01')*

The attribute technician_ID has to be unique.

Technicians can change their password through the desktop application, providing their current password and the new one. The application must only update the password if the correct current password is provided.

In the hospital/clinic there can be several laboratories, so each technician has an associated *laboratory_code* which he belongs to (assuming that each technician can work in just one laboratory).

Treatments

Doctors can prescribe treatments to patients, using the desktop application. Each treatment contains the following information:

1. *typology ((e.g. “Cardiological treatment”, or “Orthopedic treatment”, ..)*
2. *prescribed_by*
3. *prescribed_to*
4. *date*
5. *start_date*
6. *end_date*
7. *description*

The typology property contains information about the treatment medical sector.
The property *date* contains the timestamp of the moment when the treatment has been written.
The properties *start_date* and *end_date* specify the period in which the treatment has to be followed and the description reports the treatment details and modalities (e.g. Dosage adjustment of the antidiabetic drug). Two treatments of the same typology for the same patient that overlap are not admitted.
Different doctor can prescribe several treatments to each patient.

The treatment history of a patient can be visualized by the patient or a doctor.

Profiles

Doctors can make several profiles for a patient, through the desktop application. A profile contains a list of recommended value of some specific physiological parameters. The profiles are used by the desktop application to automatically check if a new measurement of a patient respects the recommended value. (This automatic task can be performed by a daemon for example).
Each profiles contains the following information:

1. *typology ((e.g. “Cardiological profile”, or “Orthopedic profile”, ..)*
2. *prepared_by*
3. *prepared_to*
4. *date*
5. *description*
6. *A sequence of (* physiological parameters, min_recommended_value , max_recommended_value, *measurement_unit*) quadruples

The *typology* property contains information about the profile medical sector.
A patient can have only one profile for a specific typology, so two profiles of the same typology for the same patient are not admitted.
The field *date* contains the timestamp of the moment when the profile has been prepared or modified.
The *description* reports the profile details and notes.

A patient can visualize his profiles through his mobile application.
A doctor can visualize the profiles of a given patient, too.

Warnings

A process of the application automatically check if a new measurement of a patient respects the recommended values specified by the patient profiles.
If a recommended value is not respected, a warning is automatically generated and stored into the database. In this way, a doctor or a lab technician can visualize the list of warnings of a given patient in a given period in order to have a glimpse about the patient problems.

Each warning contains the following information:

1. *patient_profile_associated*
2. *timestamp*
3. *related_measurement*

The properties *related_measurement* and *patient_profile_associated* contain the identifiers of the measurement and patient profile that have generated the warning.
The timestamp property indicates when the warning has been produced.

Reports

Laboratory technicians can periodically write reports about patient measurements. Each report is related to a specific patient and contains the following information:

1. *typology (e.g. "Cardiological report", or "Orthopedic report", ..)*
2. *written_by*
3. *related_to*
4. *date*
5. *start_date*
6. *end_date*
7. *notes*
8. *A sequence of (* physiological parameters, computed value, measurement_unit) triples *(e.g. (average_blood_pressure ,computed_value,unit), (max_blood_sugar, computed_value,unit), ..)*

When a *laboratory_technician* writes a report, he computes some aggregate values based on the patient measurements in a given period and he saves them into the report, adding eventual notes.
The property *date* contains the timestamp of the moment when the report has been written.
The properties start_date and end_date specify the period of the measurements analyzed to compute the aggregate values.

A patient can visualize the history of his reports in a given period.
A doctor can visualize the reports of a given patient in a given period, too.

Patients

The clinic/hospital create an account for each patient through the desktop application. Each account has the following information:

1. *patient_health_code*
2. *name*
3. *surname*
4. *date_of_birth (represented by an integer value following this schema: YYYYMMDD. For example "19951020" means 20 September 1995)*
5. *telephone_number*
6. *home_address*

The patient_health_code is a unique identifier.
A patient can associate a new sensor to his account through the mobile application (e.g. scanning a QR code).
A patient can visualize the history of his treatments and reports as already specified.

Sensors

Sensors are used for the patient monitoring. Each sensor has the following information

1. *sensor_serial_number*
2. *typology (E.G. "temperature sensor" or "motion sensor" or "blood glucose sensor"...)*
3. *description*
4. *associated_to*

All *sensor_serial_number* codes have to be unique.
The property *description* contains a short explanation about the sensor and how it works.
The property *associated_to* contains the patient ID who the sensor is monitoring.

Measurements

Periodically a sensor makes a measurement that has the following information:

1. *sensor_serial_number*
2. *timestamp*
3. *measurement_unit*
4. *value*

The field *sensor_serial_number* denotes the sensor which has made the measurement.
Timestamp obviously denotes the moment when the measurement is made.

SYSTEM REQUIREMENTS

The deployment of the application and the test database should involve a cluster of machines.

The infrastructure team has agreed to the following Availability requirements: a Strong Consistency for 100% of the data must be provided when one node is down.

The product team has agreed to the following consistency requirements:

- All reading operations performed by a doctor or a lab technician about patient data (profiles, treatments, sensors, measurements, warnings) must be strongly consistent.
- All reading operations performed by a patient about his data (profiles, list of sensors associated) may be eventually consistent unless reading his treatments that must be strongly consistent.
- Reading information about doctor, lab technician and patient accounts must be strongly consistent. (in order to guarantee a correct login process)

TECHNOLOGY

From the application requirements, it follows that a typical workload mixes both write and read operations. For this reasons, we propose to rely on Cassandra for data storage and management. Indeed, it is well known that Cassandra provides very good performance for both kinds of operations and allows handling dynamic columns, which are very useful for the flexible data schema of report, profile and warning entities.

DATASET

We want to rely on a synthetically generated dataset for instance generation, here you find some examples:

Doctors

doctor_ID	password	name	surname	date_of_birth	specialization
00000001	Pippo123	Carlo	Verde	19751211	Cardiologist
00000002	qwerty	Martim	Rivera	19701015	Orthopedist
00000003	Asd123	Luca	Rosso	18690113	Cardiologist
00000004	1234	Stefano	Marelli	18600504	Oculist

Laboratory Technicians

technician_ID	password	name	surname	date_of_birth	laboratory_code
00000001	11111	John	Masi	19820606	BIOLAB01
00000002	abc123	Stefano	Capelli	19791001	CHILAB03
00000003	monkey	Andres	Martearena	19770312	BIOLAB02

Treatments

typology	prescribed_by	prescribed_to	date	start_date	end_date	description
Cardiological	00000001	CMRNDR90P13D969L	20171110	20171111	20180220	A description
Orthopedic	00000002	ARNLCU87S23D969Z	20170623	20170624	20180624	A description
Orthopedic	00000002	BRMLSS68R18D969R	20161130	20161201	20170210	A description
Ophthalmological	00000004	TRLMRC70P22D969L	20161201	20161202	20170506	A description

Profiles (with dynamic columns)

typology	prepared_by	prepared_to	date	description	blood_pressure			glucose_level		
					min	max	unit	min	max	unit
Cardiological	00000001	ARNLCU87S23D969Z	20151210	A description	80	120	mmHg	4.0	5.9	mmol/L
Orthopedic	00000002	TRLMRC70P22D969L	20151021	A description	84	110	mmHg			

Reports (with dynamic columns)

typology	written_by	related_to	date	start_date	end_date	notes	Avg_blood_presssure		Max_glucose_level	
							value	unit	Value	unit
Cardiological	00000001	ARNLCU87S23D969Z	20170124	20170101	20170201	notes	100	mmHg	5.5	mmol/L
Orthopedic	00000002	ARNLCU87S23D969Z	20170720	20170201	20170301	notes	115	mmHg		

Patients

patient_health_code	name	surname	date_of_birth	telephone_number	home_address
ARNLCU87S23D969Z	Arena	Luca	19781201	3470433456	home address
CMRNDR90P13D969L	Camera	Andrea	19730716	3485440319	home address
BRMLSS68R18D969R	Brema	Alessio	18790713	3334669510	home address
TRLMRC70P22D969L	Terano	Marco	18610504	3479511231	home address

Sensors

sensor_serial_number	typology	description	associated_to
0001	glucose_sensor	A short description	ARNLCU87S23D969Z
0002	pression_sensor	A short description	TRLMRC70P22D969L

Measurements

sensor_serial_number	timestamp	measurement_unit	value
0001	2015-11-12 T 10:30 UTC	mmol/L	5.5
0002	2016-O1-10 T 15:40 UTC	mmHg	110

Warning

Patient_profile_ID **Measurement_ID**

patient_health_code	profile_typology	Warning_timestamp	sensor_serial_number	Measurement_timestamp
ARNLCU87S23D969Z	Cardiological	2015-11-12 T 10:32 UTC	0001	2015-11-12 T 10:30 UTC
ARNLCU87S23D969Z	Orthopedic	2015-11-12 T 10:33 UTC	0002	2015-11-12 T 10:30 UTC